

THE DISTRIBUTED COMMON GROUND SYSTEM–ARMY USER INTERFACE

A thesis presented to the Faculty of the U.S. Army
Command and General Staff College in partial
fulfillment of the requirements for the
degree

MASTER OF MILITARY ART AND SCIENCE
General Studies

by

MATTHEW J. OSTHOFF, MAJOR, U.S. ARMY RESERVES
M.P.A., University of Illinois at Springfield, Springfield, Illinois, 2014

Fort Leavenworth, Kansas
2015

Approved for public release; distribution is unlimited.

REPORT DOCUMENTATION PAGE				Form Approved OMB No. 0704-0188	
Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing this collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden to Department of Defense, Washington Headquarters Services, Directorate for Information Operations and Reports (0704-0188), 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302. Respondents should be aware that notwithstanding any other provision of law, no person shall be subject to any penalty for failing to comply with a collection of information if it does not display a currently valid OMB control number. PLEASE DO NOT RETURN YOUR FORM TO THE ABOVE ADDRESS.					
1. REPORT DATE (DD-MM-YYYY) 07-07-2014		2. REPORT TYPE Master's Thesis		3. DATES COVERED (From - To) AUG 2014 – JUN 2015	
4. TITLE AND SUBTITLE The Distributed Common Ground System–Army User Interface				5a. CONTRACT NUMBER	
				5b. GRANT NUMBER	
				5c. PROGRAM ELEMENT NUMBER	
6. AUTHOR(S) Major Matthew J. Osthoff				5d. PROJECT NUMBER	
				5e. TASK NUMBER	
				5f. WORK UNIT NUMBER	
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) U.S. Army Command and General Staff College ATTN: ATZL-SWD-GD Fort Leavenworth, KS 66027-2301				8. PERFORMING ORG REPORT NUMBER	
9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)				10. SPONSOR/MONITOR'S ACRONYM(S)	
				11. SPONSOR/MONITOR'S REPORT NUMBER(S)	
12. DISTRIBUTION / AVAILABILITY STATEMENT Approved for Public Release; Distribution is Unlimited					
13. SUPPLEMENTARY NOTES					
14. ABSTRACT The Distributed Common Ground System–Army (DCGS-A) is an intelligence system that was designed for intelligence analysts to access full-spectrum intelligence and produce analytical intelligence products. In recent years, the system has received scrutiny from members of the United States Congress due to its perceived lack of effectiveness. Popular opinion of the DCGS-A user interface within the military is it is unfriendly to use and not intuitive. This study focuses on the DCGS-A user interface to determine if an intuitive user interface will improve ease-of-use and functionality. The literature review covers a study of the characteristics software failures, select case studies to illustrate real-world applications of software development, an introduction to software design models, and an overview of the of the DCGS-A training design. This study used a survey methodology the researcher applied to an online survey to facilitate questions about the DCGS-A user interface. A group of respondents from the Command and General Staff Officers' Course furnished both qualitative and quantitative data. Three key conclusions indicated some hardware and bandwidth issues associated with DCGS-A, the system is not easy to use, and the training program is robust, but training alone does not make the system easier to use.					
15. SUBJECT TERMS Distributed Common Ground System-Army, DCGS-A, DCGS-A training, user interface, software failure, ease-of-use, intuitive, software, software development, software design,					
16. SECURITY CLASSIFICATION OF: Unclassified			17. LIMITATION OF ABSTRACT (U)	18. NUMBER OF PAGES 110	19a. NAME OF RESPONSIBLE PERSON Matthew J. Osthoff
a. REPORT (U)	b. ABSTRACT (U)	c. THIS PAGE (U)			19b. PHONE NUMBER (include area code) 708-768-2261

Standard Form 298 (Rev. 8-98)
Prescribed by ANSI Std. Z39.18

MASTER OF MILITARY ART AND SCIENCE

THESIS APPROVAL PAGE

Name of Candidate: Major Matthew J. Osthoff

Thesis Title: The Distributed Common Ground System–Army User Interface

Approved by:

_____, Thesis Committee Chair
Lieutenant Colonel, Candy Smith, M.A.

_____, Member
Donald B. Connelly, Ph.D.

_____, Member
William K. McCurry, M.B.A.

Accepted this 12th day of June 2015 by:

_____, Director, Graduate Degree Programs
Robert F. Baumann, Ph.D.

The opinions and conclusions expressed herein are those of the student author and do not necessarily represent the views of the U.S. Army Command and General Staff College or any other governmental agency. (References to this study should include the foregoing statement.)

ABSTRACT

THE DISTRIBUTED COMMON GROUND SYSTEM–ARMY USER INTERFACE, by Major Matthew J. Osthoff, 110 pages.

The Distributed Common Ground System–Army (DCGS-A) is an intelligence system that was designed for intelligence analysts to access full-spectrum intelligence and produce analytical intelligence products. In recent years, the system has received scrutiny from members of the United States Congress due to its perceived lack of effectiveness. Popular opinion of the DCGS-A user interface within the military is it is unfriendly to use and not intuitive. This study focuses on the DCGS-A user interface to determine if an intuitive user interface will improve ease-of-use and functionality. The literature review covers a study of the characteristics software failures, select case studies to illustrate real-world applications of software development, an introduction to software design models, and an overview of the of the DCGS-A training design. This study used a survey methodology the researcher applied to an online survey to facilitate questions about the DCGS-A user interface. A group of respondents from the Command and General Staff Officers' Course furnished both qualitative and quantitative data. Three key conclusions indicated some hardware and bandwidth issues associated with DCGS-A, the system is not easy to use, and the training program is robust, but training alone does not make the system easier to use.

ACKNOWLEDGMENTS

This thesis could not have been created without the support of my family. First and foremost, I want to thank my wife for her patience, support, and love. Creating this thesis has been a labor of both love and hate for the two of us. Her unconditional support gave me the inspiration needed to meet deadlines and complete all of my requirements. I want to send a special thank you to the DCGS-A program manager and his team at Fort Huachuca for providing me with special insights into the DCGS-A system and future direction the program will take. These insights allowed me to tailor my research towards areas that will give direction for future research topics. Dr. Maria Clark was instrumental in giving me the direction I needed to complete chapter 3. She provided me the path to complete and distribute the online survey and took time away from her schedule to help finalize the survey. I want to thank Dr. David Bitters for taking time out of his busy day to help me re-understand quantitative data analysis. Without his help, I would not have been able to properly analyze much of the data from the survey results. I also want to give thanks to the members of my MMAS committee. They gave me valuable feedback that allowed me to critically think about my topic. Lastly, I want to thank LTC Candy Smith for her advice, direction, and support throughout the process of creating this paper. Without her patience and feedback, I would not have been able to complete this research project.

TABLE OF CONTENTS

	Page
MASTER OF MILITARY ART AND SCIENCE THESIS APPROVAL PAGE	iii
ABSTRACT.....	iv
ACKNOWLEDGMENTS	v
TABLE OF CONTENTS.....	vi
ACRONYMS.....	ix
ILLUSTRATIONS	x
TABLES	xi
CHAPTER 1 INTRODUCTION	1
Primary Research Question	8
Secondary Research Question	8
Assumptions.....	8
Definitions	8
Limitations	10
Delimitations.....	10
Summary.....	10
CHAPTER 2 LITERATURE REVIEW	12
Overview.....	12
Software Failure Characteristics	13
Software Failures	18
Software Achievements	20
Software Design Model	22
User Interfaces	24
DCGS-A Training Design	28
Summary.....	33
CHAPTER 3 RESEARCH METHODOLOGY	35
Overview.....	35
Survey Design.....	36
Online Survey	37
Survey Instrument.....	37
Quantitative.....	38

Qualitative.....	39
Demographics	40
Population and Sampling.....	40
Margin of Error and Confidence Level.....	42
Survey Administration.....	43
Protection of Human Rights	44
Data Analysis.....	44
Quantitative.....	44
Qualitative.....	45
Summary.....	46
CHAPTER 4 SURVEY RESULTS	47
Overview.....	47
Research Questions.....	48
Survey	48
Survey Participants	50
Research Question 1: What other issues affect the operability of DCGS-A?	54
Open-Ended Question: How does bandwidth affect functionality of DCGS-A?	54
Open-Ended Question: How does hardware problems affect the functionality of DCGS-A?.....	55
Research Question 2: Does the user interface of the current version of DCGS-A detract from the capability to analyze and disseminate intelligence in a timely fashion?.....	55
DCGS-A Tools.....	55
Statistical Analysis.....	56
Workspace Integration	59
Open-Ended Question: What combined components of DCGS-A would make DCGS-A easier to use?	60
Open-Ended Question: What combined components of DCGS-A would make DCGS-A more functional?	61
Opinions on the DCGS-A User Interface	61
Open-Ended Question: How could the DCGS-A user interface be improved?.....	62
Open-Ended Question: What would make the DCGS-A user interface more functional?.....	62
Research Question 3: Does the amount of time spent using and-or training on DCGS-A hinder the user's ability to operate the system?	62
Training on DCGS-A.....	62
Open-Ended Question: Comments on DCGS-A training.....	64
Open-Ended Question: How is your ability to operate DCGS-A affected by the amount of time between uses?	65
Survey Comments.....	65
Summary.....	66

CHAPTER 5 CONCLUSION AND RECOMMENDATIONS	68
Research Question 1: What other issues affect the operability of DCGS-A?	69
Discussion	69
DCGS-A Bandwidth Issues	69
DCGS-A Hardware Issues	69
Recommendations	70
Research Question 2: Does the user interface of the current version of DCGS-A detract from the capability to analyze and disseminate intelligence in a timely fashion?	71
Discussion	71
DCGS-A Tools	71
Workspace Integration	72
DCGS-A User Interface Opinions	73
Recommendations	73
Research Question #3: Does the amount of time spent using and-or training on DCGS-A hinder the user's ability to operate the system?	74
Discussion	74
DCGS-A Training	75
Recommendations	75
Comments on DCGS-A	77
Discussion	77
Projecting DCGS-A into the Future	77
Recommendations	78
Summary and Implications	78
GLOSSARY	80
APPENDIX A ONLINE SURVEY	81
BIBLIOGRAPHY	96

ACRONYMS

AAR	After Action Review
CGSC	Command and General Staff College
CGSOC	Command and General Staff Officers' Course
DCGS-A	Distributed Common Ground System–Army
HUMINT	Human Intelligence
MDSD	Model Driven Systems Development
MOS	Military Occupation Specialty
RAF	Regionally Aligned Forces
TSP	Training Support Package
UI	User Interface
USAR	United States Army Reserves

ILLUSTRATIONS

	Page
Figure 1. Adaptive UI and Human Problems Mode.....	26

TABLES

	Page
Table 1. Qualifying Question.....	52
Table 2. Population Demographics.....	52
Table 3. Using DCGS-A During Times Other Than For Training.....	53
Table 4. Experience Using DCGS-A	53
Table 5. Experience Using DCGS-A While Deployed.....	53
Table 6. Other Issues Affecting DCGS-A Operability	54
Table 7. DCGS-A Tools.....	57
Table 8. Ease of Use with an Integrated DCGS-A Workspace	59
Table 9. A More Functional DCGS-A Workspace	60
Table 10. Easy to Use DCGS-A Interface	61
Table 11. Time Spent Using DCGS-A	63
Table 12. Time Spent Training on DCGS-A	63
Table 13. DCGS-A Training and Skill Retention.....	64

CHAPTER 1

INTRODUCTION

Overview

An interface is humane if it is responsive to human needs and considerate of human frailties. If you want to create a humane interface, you must have an understanding of the relevant information on how both humans and machines operate.

— Jef Raskin, *The Humane Interface: New Directions for Designing Interactive Systems*

New-age electronic technology tailored to the military's needs has been around for quite some time. Over the last few decades, the United States government has begun to incorporate commercial off-the-shelf (COTS) technology into military equipment because of its ease-of use, maintenance, and the agile ability to upgrade quickly.¹ Military computer programs that have blatantly failed in the past have cost the U.S. government billions of dollars² and has driven military leaders into making cost-effective

¹ John Scott et al., "Open Technology Development (OTD): Lessons Learned and Best Practices for Military Software," *2011 OTD Lessons Learned*, May 16, 2011, accessed September 12, 2014, <http://mil-oss.org/resources/books/open-technology-development>.

² Mark Flatten, "Army Can't Track Spending on \$4.3b System to Track Spending, IG Finds," *Washington Examiner*, September 4, 2014, accessed November 1, 2014, <http://www.washingtonexaminer.com/army-cant-track-spending-on-4.3b-system-to-track-spending-ig-finds/article/2552808>; Tony Capaccio, "Senators to Probe Air Force's #1 Billion Failed Software," *Bloomberg*, January 24, 2013, accessed November 1, 2014, <http://www.bloomberg.com/news/2013-01-24/senators-to-probe-air-force-s-1-billion-failed-software.html>; Randall Stross, "Billion Dollar Flop: Air Force Stumbles on Software Plan," *The New York Times*, December 8, 2012, accessed November 1, 2014, <http://www.nytimes.com/2012/12/09/technology/air-force-stumbles-over-software-modernization-project.html>.

decisions that lead to private-industry solutions.³ The Distributed Common Ground System–Army (DCGS-A) is one such military computer program that has undergone recent scrutiny by leaders in Congress. DCGS-A is the Army’s intelligence system that supports intelligence analysts ability to gather, analyze and share significant amounts of information pulled into a common environment, and enhances Soldier situational awareness and improves the Commander’s ability to protect the force.⁴ In 2007, the Army declared DCGS-A as a program of record, meaning DCGS-A was slated for acquisition and received the allocated budget requests submitted by Army leaders. During the acquisition process, Army leaders used a significant number of both public and private industry partners to contribute to the development of DCGS-A. Today, the system uses more than forty full-time industry partners. The system functions by drawing on more than 600 sources of information from Global Hawk drones and GPS satellites to ground sensors and biometric scanners. It uses a mix of military and commercial software applications, including Google Earth made by Google, Inc. and i2 Analyst’s Notebook made by IBM Corporation.⁵ DCGS-A operates across all echelons from the company-level intelligence support team (CoIST) to the Army Service Component Command (ASCC). It operates on all security and network domains including coalition networks, non-secure internet protocol router network (NIPRNet), secret internet protocol router

³ Scott et al.

⁴ Distributed Common Ground System-Army, “About DCGS-A,” last modified May 4, 2014, accessed October 6, 2014, http://dcgsa.apg.army.mil/about_dcgsa/.

⁵ Brendan McGarry, “Army Units in Afghanistan Slam Intel System,” DoD Buzz, February 6, 2014, accessed September 14, 2014, <http://www.dodbuzz.com/2014/02/06/army-units-in-afghanistan-slam-intel-system/>.

network (SIPRNet), and the joint worldwide intelligence communications system (JWICS). It provides the Operating Force (OF) conducting decisive action operations and through all phases of training and deployment with a fully compatible Intelligence, Surveillance and Reconnaissance (ISR) ground processing system capable of supporting each security and network domain.

The term interface indicates many different contact points and exchanges between different programs and data layers in a computer, between different machines (a network of computers), between humans and machines (such as graphical user interfaces (GUIs)), and as a mediator between humans.⁶ The Army designed the DCGS-A UI as a conglomerate of piecemealed programs to handle the numerous applications, tools, and widgets that users employ to produce, analyze, and manipulate intelligence.

A user interface's ease of use is a common concern among end users because it is a mitigating factor for productivity, and profits. It is important to note that there is no connection between the ease of use and the speed of learning.⁷ Intuitive interfaces increase the speed of learning, whereas ease of use increases functionality. This research will explore UI intuitiveness and ease of use in chapter 2 of this study. The DCGS-A ease-of-use issue is a common theme throughout many outlets, including Army after action reviews (AARs), media, and congressional testimony. The Army has gone through several software upgrades in an attempt to provide software solutions from issues brought

⁶ Christian U. Andersen and Soren B. Pold, *Interface Criticism: Aesthetics Beyond Buttons* (Oakville, CT: The David Brown Book Company, 2011), 9.

⁷ Jef Raskin, *The Humane Interface: New Directions for Designing Interactive Systems* (New York, NY: ACM Press, 2000), 203.

up in official Army AARs and lessons learned. The AARs and lessons learned present clear issues to Army leadership about the problems facing DCGS-A. Recommendations to have more in-depth training sessions on DCGS-A⁸ were submitted in an AAR comment by the 34th Infantry Division. Although this 2010 AAR comment is five years old, it indicates a need for DCGS-A training enterprise reform. The Center for Army Lessons Learned indicated that although DCGS-A has powerful analytical tools, is not easy to learn and is not meant for use by the ground warfighters.⁹ This AAR comment focuses on not only the system's ease-of-use, but identifies key challenges in training soldiers to meet the training standard. Unofficial comments by DCGS-A users claim the system is clunky not easy to use. They prefer to use systems like Palantir that has a great UI and is less complicated to use.¹⁰ This comment on DCGS-A explains one of the core concerns this study will attempt to answer. Modern intuitive and easy to use software programs have become more popular in recent times and are likely the reason why these types of comments are surfacing about DCGS-A. These concerns about DCGS-A's ease-of-use reinforce the idea the UI needs to be redesigned to become an easier to use system.

The Army has attempted to improve DCGS-A by fielding upgrades and releasing new versions of DCGS-A software to the end users. The current release of DCGS-A is

⁸ 34th Infantry Division, *After Action Report, Predeployment*, vol. 1 (Rosemount, MN: 34th Infantry Division, July 2010).

⁹ MAJ Robert Gregor, "Intelligence, Intelligence Trends," *Bronco Team Newsletter: Tactics, Techniques, and Procedures*, no. 10-21 (January 2010): 13.

¹⁰ Kristina Wong, "Army Defends Intel System Ahead of House Hearing," *The Hill*, March 25, 2014, accessed January 27, 2014, <https://thehill.com/policy/defense/201604-army-defends-intel-program-ahead-of-house-hearing>.

version 3, increment 1, release 2 (hunte). Version 3, increment 1, release 1 (griffin) is a previous release that focused on leveraging quick-reaction efforts for program of record, and ease of use. It provided intelligence community alignment, geospatial intelligence exploitation and foundation capability, full motion video ingestion, exploitation, and dissemination, and all-source analysis leveraging over 700 data sources. Release 2 was a quick release update that the Army deployed as a high side delivery to support sensitive compartmented information (SCI) operations. This release had human intelligence (HUMINT) data interoperability, and provided expanded ease of use enhancements. The unsolved issues that software upgrades did not resolve and has steered commanders to other options. DCGS-A was intended to provide commanders with a common operating picture of the enemy, but have instead become a complicated and unreliable system that has steered commanders into using other program such as Microsoft PowerPoint and Palantir Technology Incorporated's Palantir.¹¹ The Army intends the use of software such as Palantir as a temporary solution while DCGS-A developers continue to plan future upgrades.

In 2011, Congress asked Army leaders and the Secretary of Defense why DCGS-A has constantly gone over budget, is five-years behind schedule, fails to meet the needs of soldiers, it is unable to perform simple tasks, and frequently crashes.¹² An explanation fell silent for quite some time. General Raymond Odierno, Army Chief of Staff, and Heidi Shyu, Chief of Army acquisition testified in front of the United States House of

¹¹ McGarry,

¹² *Testimony by Secretary Penetta and Gen. Dempsey Before the House Armed Services Committee*, 112th Cong., 1st sess, October 13, 2011.

Representatives Armed Services Committee in 2013, defending DCGS-A. General Odierno got into a heated debate with Representative Duncan Hunter over the inadequacies of DCGS-A. General Odierno poignantly responded, “I’m tired of somebody telling me that I don’t care about our soldiers, that we don’t respond. Everybody on my staff cares about it, and they do all they can to help.”¹³ The publicly-made 2010 memorandum authored by Lieutenant General Michael Flynn that fueled DCGS-A public debate, outlined combat-related DCGS-A issues forced the Army to make rapid acquisition requests to fix the DCGS-A problems.¹⁴ LTG Flynn requested a “theater-wide, web-based analytical platform” resembling Palantir’s UI.¹⁵ The Army continues to look ahead to the future to improve DCGS-A. Increment 2 is in developmental stages and the design is planning to expand on the capabilities provided by increment 1. Increment 2 will add capabilities at the Army and below echelons while developing new, enhanced, and leap-ahead ISR capabilities to align with the intelligence community and the Army’s contemporary operating environment. Developers for increment 2 plan to issue a series of requests for information (RFIs) to develop a competitive multiple-industry event that will define the acquisition strategy approach. Future DCGS-A updates plan to build upon emerging technologies such as an artificial

¹³ Austin Wright, “Duncan Hunter Battles Army Over Intel-Gather System,” Politico, May 1, 2013, accessed September 12, 2014, <http://www.politico.com/story/2013/05/duncan-hunter-battles-army-over-intel-gathering-system-90796.html>.

¹⁴ Lieutenant General Michael Flynn, Memorandum, *Advance Analytical Capability Joint Urgent Operational Need Statement*, July 2, 2010,

¹⁵ Ken Dilanian, “Top Brass Defend Troubled Intelligence System,” Standard Examiner, July 10, 2014, accessed January 30, 2014, <http://www.standard.net/National/2014/07/10/Top-Army-brass-defend-troubled-intelligence-system>.

intelligence system capability, additional exploitation tools, and capabilities on the explosive growth in unstructured data (social networks and smart devices). It is unclear the components of DCGS-A that will be improved. Heidi Shyu admits that Palantir has a good product, and Palantir could possibly be integrated into future DCGS-A upgrades.¹⁶ However, part of the acquisition process is opening bids to the industry where anyone can have the opportunity to submit a great idea.

This study will focus on the DCGS-A user interface (UI) that uses both government and COTS programs. The exploration of UI models will include the idea of improving interface models to reduce training time and improve skill retention over time through intuitive functionality. In this context, intuitive functionality is software that operates just like some other type of software or method the user is familiar. Intuitive in this context means habitual, whereas over time, the software becomes increasingly easier to use. The study will explore private sector computer software models that integrate human psychological aspects and provide an in-depth analysis on the successes and failures of both private sector and certain government computer software. It is important to understand the difference in motivations between public and private sector software development. In the private sector, businesses are accountable to shareholders and their primary motivation is profit, whereas in the public sector, organizations are ultimately accountable to governmental bodies and taxpayers. Deficiencies of the current UI on DCGS-A will be analyzed as well as survey results in order to submit a recommendation that will theoretically provide the building blocks for a UI solution to DCGS-A.

¹⁶ Wong.

Primary Research Question

How does the UI within DCGS-A affect operability of the system?

Secondary Research Question

What other issues affect the operability of DCGS-A?

Does the UI of the current version of DCGS-A detract from the capability to analyze and disseminate intelligence in a timely fashion?

If yes, what part(s) of the UI hinder the ability?

Does the amount of time spent using and-or training on DCGS-A hinder the user's ability to operate the system?

Can a change in the UI allow easier user functionality with less training time?

Have there been successful applications of redesigned UIs?

If yes, what design models do the private sector use?

Are there any considerations for psychological aspects?

What types of changes to the DCGS-A UI would benefit the warfighter?

Assumptions

It is assumed that the sample population surveyed and the responding participants represent the larger Army population of DCGS-A operators. The survey participants will answer the questions honestly.

Definitions

Distributed Common Ground System–Army: The Distributed Common Ground System – Army is the Army's Intelligence component that gathers intelligence spanning all echelons from space to mud. DCGS-A enables decision makers to save lives right now

by gathering, analyzing and sharing intelligence information into a common system. This enterprise utilizes intelligence information and open source technology (such as Amazon, Google and Twitter) to create a scalable environment for collaboration and intelligence production. Overall, it has 600 sources of information.¹⁷

Private sector: The area of the nation's economy under private rather than governmental control.¹⁸

Public sector: The public sector consists of governments and all publicly controlled or publicly funded agencies, enterprises, and other entities that deliver public programs, goods, or services.¹⁹

User interface: A graphical display of information displayed on a computer screen based on metaphors of documents, files, folders, and applications that are used for various forms of manipulation.²⁰

¹⁷ Distributed Common Ground System-Army, "About DCGS-A."

¹⁸ Dictionary.com, "Private Sector," accessed October 8, 2014, [http://dictionary.reference.com/browse/private sector](http://dictionary.reference.com/browse/private%20sector).

¹⁹ Sikhungo Dube and Daniela Danescu, "Supplemental Guidance: Public Sector Definition," The Institute of Internal Auditors (Altamonte Springs, FL: Global Headquarters, December 2011), accessed August 29, 2014, <https://na.theiia.org/standards-guidance/Public%20Documents/Public%20Sector%20Definition.pdf>.

²⁰ Catherine Plaisant and Ben Shneiderman, "Personal Role Management: Overview and a Design Study of Email for University Students," in *Beyond the Desktop Metaphor: Designing Integrated Digital Work Environments*, ed. Victor Katelinin and Mary Czerwinski (Cambridge, MA: The MIT Press, 2007), 144.

Limitations

Time constrain is a major limitation of this study which will in turn limit the amount of qualitative and quantitative data collected. A limited and select number of soldiers with specific military occupation specialties (MOS) serving in the regular Army, Army Reserves, and Army National Guard within U.S. Army Training and Doctrine Command (TRADOC) will receive a survey to answer questions about the DCGS-A UI. In order to ensure sufficient samples of data are collected, this study will require a minimal number of officers, warrant officers, and enlisted participants within this population to respond to the survey. A number of respondents less than the minimal number of participants expected may not be representative of the population of DCGS-A users. Without sufficient participation in the survey, there may not be sufficient data to explain the research question this paper focuses on. A low response to the survey will threaten external validity.

Delimitations

The surveys will be restricted to soldiers assigned to schools within TRADOC who are or have been operators of DCGS-A with any amount of experience and are members of the active, Reserve and National Guard components of the United States Army. The researcher will place an additional constraint on time. Survey collection will be limited to the period of March 15 to April 15, 2015.

Summary

Chapter 1 identified the current issues facing the DCGS-A enterprise, and the topic of this study. Top leaders in the Army continue to defend the system as being a

robust system that provides intelligence better than any system before it. Voices throughout the DCGS-A community expressed their concerns over their perceived problems with the system. The researcher identified the direction the study will take and the areas where it will concentrate. The researcher applied particular constraints, assumptions, and limitations due to the survey methodology used to collect data.

The next chapter of this study will explore the common characteristics of software failures and describes the intricacies of software design models used in modern intuitive software programs. The review examines a few case studies to provide valuable insight into some reasons why software fails and why some become successful. A portion of the review focuses on the design of the DCGS-A training program to provide an overview of the training enterprise in its current state. The review provides a better understanding of common pitfalls and mistakes that lead to software failure, and informs on emerging UI models that are the foundation for intuitive software programs.

CHAPTER 2

LITERATURE REVIEW

Overview

This chapter provides an understanding of the common characteristics of software that fails to meet end-user expectations. The researcher provides an overview of software failure case studies, software design models, and an overview of user interface (UI) design models in order to understand what causes software to fail to meet end-user expectations. The professional sources used in this chapter are well-known experts in the computer programming industry and have published books, professional journal articles, and in many cases, have advanced post-graduate degrees. The theories and ideas these professionals presented in their works and revealed in this chapter can be applied to the development of new systems and user interfaces.

The case studies of software system failures at the IRS, Hershey Corporation, and McDonald's Corporation provide insight to some of the common mistakes that causes software development to fail. Focusing on both private and public sector software systems will give good insight into systemic issues that plague software failures and can provide good resource into what works well with software programs. A detailed look into the failures and success of software systems will reveal four broad categories of information technology (IT) blunders and accomplishments: people, process, product, and technology.²¹ Software design models are the basis upon how software is developed.

²¹ R. Ryan Nelson, "IT Project Management: Infamous Failures, Classic Mistakes, and Best Practices," *MIS Quarterly Executive* 6, no. 2 (June 2007): 67-78.

Exploring the two most widely used software development models will provide a more in-depth understanding into the processes of designing system architectures. Above all, software developers consider the UI to be one of the key components of software applications since it connects their end-users to the functionality.²² Well-engineered, well-managed, and robust software applications are prone to failure due to a weak UI. The exploration of the key characteristics of software failure, and case studies offer classic examples that developers should avoid when developing software. The researcher presents software development models to provide a concise understanding of the most modern techniques for building innovative and intuitive software. Lastly, the researcher explores the current version of DCGS-A and its training strategy to provide significant insight into the DCGS-A training design and learning model.

Software Failure Characteristics

Software development is a labor-intensive, intellectually demanding creative activity. There are many reasons why software projects fail. In most instances, humans are to blame for the inadequacies of software.²³ Lack of adherence to schedules, improper program management, and inadequate resource management are a few of the most popular negative human factors that can lead to software failure include. Project abandonment inevitably leads to project failure. These are the most common factors of software failure are: (1) Having unrealistic or unarticulated project goals; (2) Inaccurate

²² Pierre Akiki et al., “Adaptive Model-Driven User Interface Development Systems,” *ACM Computing Surveys* 47, no. 1 (April 2014): 9-1 to 9-33.

²³ Carol Righi and Janice James, *User-Centered Design Stories: Real-World UCD Case Files* (San Francisco, CA: Morgan Kaufmann Publishers, 2007), 27.

estimates of needed resources; (3) Badly defined system requirements; (4) Poor reporting of the project's status; (5) Unmanaged risks; (6) Poor communication among customers, developers, and users; (7) Use of immature technology; (8) The inability to handle the project's complexity; (9) Sloppy development practices; (10) Poor project management; (11) Stakeholder politics; and (12) Commercial pressures.²⁴ The inability to implement the software to perform to the expectations of the user and the inability of software developers to produce a working or functioning system for the users are two broad categories that emerge from the common factors. R. Ryan Nelson discovered the most significant reason for software development failure was poor estimation and-or scheduling. The next nine instances (in order of most to least significant) were ineffective stakeholder management, insufficient risk management, insufficient planning, shortchanged quality assurance, weak personnel and-or team issues, insufficient project sponsorship, poor requirements determination, inattention to politics, and lack of user involvement.

Research conducted over the past thirty years has surfaced some interesting human factors that contribute to the failure of software projects. One aspect to software development failure is the expectations of key stakeholders. Stakeholders' expectations can include cost and time-of-completion estimates versus the expected benefits from the system. Cost, quality, and schedule can also become a large political stakeholder interest that can greatly affect the project outcome. Poor interaction between developers and

²⁴ Robert N. Charette, "Why Software Fails," *IEEE Spectrum*, September 5, 2005, accessed October 20, 2014, <http://spectrum.ieee.org/computing/software/why-software-fails>.

stakeholders can lead to software failure. Failures and improper oversight in the development process, systems objectives, cost, and other resources needed to produce a functioning system within expected periods to the satisfaction of stakeholders are what complicates software development and can lead to software failure.²⁵

Undetermined motivation has a large effect on the productivity and quality of software design.²⁶ A phenomenon in programming known as the deadline effect is one aspect of programmer motivation. The deadline effect is happens whenever a deadline for completing a project approaches. Effort to complete the project increases significantly as the deadline draws nearer. As activity increases, an increased risk to meet the deadline schedule increases and there is a need for more individuals to complete the product within the deadline. To reduce this risk, the project manager either pushes the schedule out farther, or adds more individuals to the project. Adding people to an already late project can add confusion to an already complicated problem, and take away from productivity from the existing team members.²⁷ Either option will result in project delays.²⁸ Project managers must properly track the work hours and time management of

²⁵ Kweku Ewusi-Mensah, *Software Development Failures* (Cambridge, MA: The MIT Press, 2003), 9.

²⁶ Barry W. Boehm, "An Experiment in Small Scale Application Software Engineering," *IEEE Transactions on Software Engineering* SE-7, no. 5 (September 1981): 482-494.

²⁷ Frederick P. Brooks Jr., "The Mythical Man-Month: After 20 Years," *IEEE Software* 12, no. 5 (September 1995): 57-60.

²⁸ Boehm, 488-490.

the employees working on the project. The software will otherwise be severely delayed or result in failure.

A term in the computer-programming world known as process is a technical methodology used to complete a software project, and the accompanying management process is what steers the project. Anil Khurana and Stephen Rosenthal identified the wasted time in the “fuzzy front end” before a project starts could be time spent to start the project, thus saving time in the overall development process.²⁹ There is also a human tendency to underestimate and produce overly optimistic schedules. Poor time estimation can set a project up for failure from the start. It undermines effective planning, shortchanges requirements determination, and-or quality assurance, among other things.³⁰ The software development effort can become a failure when the software development fails to attain any of the targeted goals.³¹ An added acquisition process must take place prior to software development within the government. Similar to this issue is the rise in outsourcing. When the project manager brings contractors into the picture, the risk of software development failure significantly rises. According to a 2007 study that looked at ninety-nine software development projects conducted by R. Ryan Nelson, 45 percent of mistakes in software projects were categorized as a process mistake, 43 percent were

²⁹ Anil Khurana and Stephen R. Rosenthal, “Integrating the Fuzzy Front End of New Product Development,” *Sloan Management Review* 38, no. 2 (Winter 1997): 103-120.

³⁰ Steve McConnell, *Software Estimation: Demystifying the Black Art* (Redmond, WA: Microsoft Press, 2006), 4.

³¹ Ewusi-Mensah, 8.

classified as people mistakes, and the remaining 12 percent were categorized as product mistakes (8 percent), or technology mistakes (4 percent).³²

After motivation, the largest influencer of productivity has been either the individual capabilities of the team members or the working relationships among the team members.³³ Differences in backgrounds and technical expertise can vary greatly amongst team members. Teams can be composed of technical personnel or developer who has technical skills and experience, but not necessarily the skills needed for the problems facing the software project. Managers that oversee the project may lack the technical background or knowledge of the software project challenges.³⁴ Unskilled or untrained managers could carry the project into a direction that might lead it to failure. To overcome these shortfalls, a software development team must communicate the crucial prerequisites, coordinate direction of the projects, and interact constantly. In addition, clearly defined roles ensure the collaborative team effort carries out each task of the project.

People issues are not the only contributing factor to software failures, however. Organizations can be at fault for failure and projects are then abandoned. Organizations must invest substantial organizational resources in a project until their criteria is met.³⁵

³² Nelson, 67-78.

³³ B. Lakhampal, "Understanding the Factors Influencing the Performance of Software Development Groups: An Exploratory Group-Level Analysis," *Information and Software Technology* 35, no. 8 (August 1993): 468-473.

³⁴ Ewusi-Mensah, 36.

³⁵ Ibid., 15.

Two classifications of software projects determine whether one is doomed to fail, or the project completes with limited functionality. A challenged software project is one that managers do not cancel, but has gone over budget and over schedule, but is completed. Challenged software will have similar characteristics to an impaired software project, but the difference is managers will eventually cancel the impaired project at some point during the developmental lifecycle. Total, substantial, and partial abandonment are classified sub-categories of software abandonment. The most severe of the three sub-categories is total abandonment because of the amount of loss associated with the abandonment.

Software Failures

Both the public and private sectors have instances of software development failure. When some of the characteristics of software failures previously described combine, projects fail. This section will explore a few instances of failed software projects as a contextual background into software project failures.

The Internal Revenue Service (IRS) launched a business systems modernization project in 1999 to upgrade the agency's IT infrastructure and more than 100 business applications. Over a course of two years, the IRS estimated billions of dollars spent on the project with no end in sight. Internal IRS reviews of the software development identified several deficiencies that actually weakened the business system and was unable to maintain proper overview on the pace of the modernization program.³⁶ This software

³⁶ United States General Accounting Office, *Testimony before the Subcommittee on Oversight, Committee on Ways and Means, House of Representatives*, 108th Cong., 2nd sess., February 12, 2004.

project highlights what can go wrong when a complex project overwhelms the management capabilities of both contractor and client.

Hershey Foods embarked on software development project to automate their back-office operations in 1996. They utilized a system called Enterprise Resource Planning (ERP). The goal of the project was to integrate the disparate legacy applications systems for handling order processing, inventory, and human resource operations at Hershey into one computing platform.³⁷ Hershey Foods estimated the project to cost between \$112 and \$115 million. Project managers scheduled software implementation for completion in four years, but an expedited time plan moved the schedule forward to thirty months. Implementation problems began to surface after the software began to roll out to the end users. It took over a year and additional funds to fix the issues to regain system integrity and normalcy. The expedited period for the software development was at fault for the failure of the software development.

In 2003, the McDonald's corporation experienced fiscal troubles and announced it would close 719 restaurants worldwide due to a loss of over \$340 million in 2002. There were \$810 million in charges the corporation said it would absorb due to those losses. Part of that loss was a \$170 million write off that was part of a planned \$1 billion software project McDonald's had started in 2001.³⁸ Software designers estimated the software project would allow management to see the amount of food sold, stocks of

³⁷ Ewusi-Mensah, 24.

³⁸ Sean Gallagher and Larry Barrett, "McDonald's: McBusted," PCMag.com, July 2, 2003, accessed November 3, 2014, <http://www.pcmag.com/article2/0,2817,1173624,00.asp>.

consumable restaurant supplies, and food temperatures, at any store at any time of the day. McDonald's planned to have the software implemented within five years after investing \$1 billion into the system. McDonald's hired Oracle to design a web-based network that was centered on the tenets of McDonald's vision of technology. Existing technology and computer systems interfered greatly with the software development project. As revenues continued to drop and project deadlines extended, costs for the project continued to climb until the project was cancelled 2002. There were between one to two hundred contractors and McDonald's employees working on the project and spent over \$170 million when the project came to a halt.³⁹ The software was difficult to use, and the technology was negatively affecting its customers and the company's profits.

Software Achievements

Successful software products have several common dimensions. The commonalities are the a well-defined product line, a common software product line architecture as the basis for each product, economies of scale, variation to realize the economies of scope, and lastly, the organization is structured and operated to facilitate building reusable assets and building products using those assets.⁴⁰ An anonymous author offers the simple suggestion of teamwork to developing successful software. The author cites that software developers are more likely to succeed with a skilled team, managerial

³⁹ Ibid.

⁴⁰ John D. McGregor et al., "Guest Editors' Introduction: Successful Software Product Line Practices." *IEEE Software* 27, no. 3 (May 2010): 16-21.

involvement and a common level of team experience that with high-tech tools.⁴¹ In his 2007 study, R. Ryan Nelson found that ineffective stakeholder management ranked second as a mistake in the software development process.

SystemForge is a software product line for developing e-commerce, content management, and other custom Web applications.⁴² Over time, the system became an unmanageable program with a multitude of applications. A redesign of the system focused on commonalities and relied on developing a common functionality within the system. Hybrid software was eventually developed and the result was customized software that met the client's needs. The upgraded software was future focused, allowing successive upgrades to be added when the client needed that capability.

Hewlett-Packard (H-P) began using the Owen software product line in 1997 to imbed its architecture into the H-P software products. Owen software began standardizing its software programming code and eventually became the standard that all inject products use. A foundational framework gives way to the software architecture that provides services like persistent storage, resource and job management, and system startup/shutdown and power orchestration.⁴³ The fact that Owen focused on standardizing key end-user functions is a reason what has made H-P products so successful.

⁴¹ "Key to Successful Software: Teamwork," *Resource* 6, no. 2 (February 1999): 4.

⁴² McGregor et al., 20.

⁴³ Ibid., 19.

Software Design Model

There are currently no accepted standards for describing process models. The vast majority of models are expressed informally, using diagrams and descriptive texts.⁴⁴

Software developers rarely develop today's software systems from scratch. The use of modeling in software development has several advantages and disadvantages. The varying types of models all intend to provide an established better understanding of the problem, communicate a clearer shared view of the problem and a solution, and formally or informally analyzing key aspects of the solution.⁴⁵ Industry defines a software model as a collection of all the artifacts that describe the overall software system. Traditional design models create a simple solution but typically result in systems that are sluggish in their response, are expensive to maintain over time, and are prone to system failure.⁴⁶ To mitigate these downfalls and provide a mechanism to fulfill the needs of the client, a model-driven systems development (MDSD) approach provides a standardized model that supports general systems engineering. It is possible to model MDSD for a business enterprise. The business model of MDSD provides specific needs and solutions for industry-specific requirements. Model-based user interface development (MBUID) is a model approach to providing specialized solutions for effectively capturing UI functionality.

⁴⁴ Ian Sommerville, "Software Process Models," *ACM Computing Surveys* 28, no. 1 (March 1996): 269-271.

⁴⁵ A. W. Brown, S. Iyengar, and S. Johnston. "A Rational Approach to Model-Driven Development," *IBM Systems Journal* 45, no. 3 (July 2006): 463-80.

⁴⁶ L. Balmelli et al., "Model-Driven Systems Development," *IBM Systems Journal* 45, no. 3 (July 2006): 569-570.

A model-driven systems development is an approach for addressing complex development challenges by dealing with complexity through abstraction.⁴⁷ It is a set of approaches where code is automatically or semi-automatically created from other abstract models. During MDSD, software development goes through a series of transformations. Each transition adds another level of detail and specificity. MDSD approaches development by breaking down the software design system into a comprehensible set of requirements. The decomposition process allows for a simplistic approach to a complex problem. The phases of transformation produce software architecture (artifacts), which includes software diagrams, informational documents, and matrices. Creating artifacts is important because they define system elements and their integration into the software.

The business model of MDSD seeks to ensure that software requirements reflect the needs of the business. This model suggests that business and technology teams should collaborate to provide a shared view of the organization's needs and priorities.⁴⁸ Software developers use system modeling platforms to define the business requirements and perfect them into software solutions. Well-defined software solutions are the basis for software architecture, analysis, UI design, and testing. The result is a system that meets business goals and requirements and delivers real business value.⁴⁹

Several generations extending from the early 1980s of MBUIDs have improved upon itself and have resulted in the three current core models that have a direct influence

⁴⁷ Balmelli et al., 571.

⁴⁸ Brown, Iyengar, and Johnston, "A Rational Approach to Model-Driven Development," 475.

⁴⁹ Ibid., 473.

on the content and appearance of a UI. The task model represents a description of the tasks that users of interactive systems can accomplish. The dialog model describes the set of actions or tasks the user can carry out within various systems, and connects the tasks with corresponding interaction elements. The interactions within the dialog model link the task and presentation models. The presentation model represents the visual, user-interactive (haptic), or auditory elements that a UI offers to its users.⁵⁰ There are several challenges to the MBUID model identified by Meixner in his 2011 study. These challenges provide ambiguity to the model but Meixner offers opinionated solutions and advice in response to the challenges. He states that solving the challenges will give the MBUID model to become widely accepted model that can provide automated generation of UI portions.

User Interfaces

Developing user interfaces has become a time-consuming and costly process. Approximately 48 percent of source code is the graphical user interface (GUI), which takes up around 45 percent of the total time for development, 50 percent of implementation, and 37 percent of maintenance time.⁵¹ A 2014 study by Pierre Akiki, et al., found that a “one size fits all” approach when developing user interfaces is unable to

⁵⁰ Gerritt Meixner et al., “Past, Present, and Future of Model-Based User Interface Development,” *I-com* 10, no. 3 (March 2011): 2-11.

⁵¹ Gerritt Meixner, Gaelle Calvary, and Joelle Coutaz, “Introduction to Model-Based User Interfaces,” W3C, January 7, 2014, accessed October 31, 2014, <http://www.w3.org/TR/mbui-intro/>.

accommodate all the cases of variability in the context of use.⁵² In many cases, this approach led to a diminished user experience. With many different types of UIs in development, adaptive UIs have become more popular in recent times due to their ability to adapt automatically to the context of use at runtime.⁵³

When developing a UI that adjusts to the individualism of the human element, it is important to ensure the UI design integrates universal psychological factors. Developers have built almost all current UIs on underpinnings that incorporate human thought and behavior.⁵⁴ An aspect of adaptive UIs is plasticity. Figure 1 is a model proposed by Pierre Akiki that describes the relationship between an adaptive UI and human problems related to adaptive UIs. Industry defines plasticity as the capacity of UIs to adapt, or to be adapted to, the content of use while preserving usability.⁵⁵ The three properties of adaptive UIs, context awareness, self-configuration, and self-optimizing demonstrate different complexity levels in software application adaptability. These categories are theoretical platforms that could promote adaptive UI behavior in future systems development.

⁵² Akiki et al., 9-1.

⁵³ Ibid.

⁵⁴ Raskin, 1-3.

⁵⁵ Joelle Coutaz, “User Interface Plasticity: Model Driven Engineering to the Limit!,” (Proceedings of the 2nd ACM SIGCHI Symposium on Engineering Interactive Computing Systems, June 2010), 1-8.

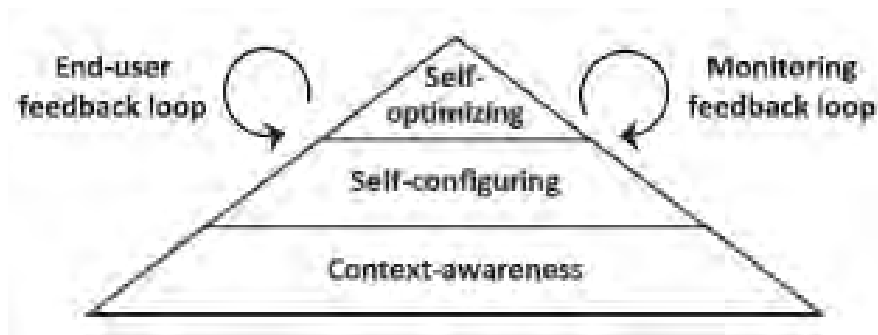


Figure 1. Adaptive UI and Human Problems Mode

Source: Pierre Akiki et al., “Adaptive Model-Driven User Interface Development Systems,” *ACM Computing Surveys* 47, no. 1 (April 2014): 9-2.

An untold number of software applications suffer from functionality problems because developers do not design their UIs to adapt to the user. Akiki suggests there are many approaches for developing adaptive UIs that target different types of software systems based on aspects such as accessibility, concurrent tasks, culture, natural context, and platform.⁵⁶ Leah Findlater suggests that in order to develop a truly adaptive UI, it takes a complex adaptive algorithm that focuses on the accuracy of adaptive predictions, behavior predictability, and the frequency that adaptability takes place within the UI.⁵⁷ The U.S. Government believes it takes time and a modern software intellectual property regime to broaden the defense industrial base and evolve common software architectures and industry-wide baselines to increase the adaptability, agility, and capacity to meet

⁵⁶ Akiki et al., 9-3.

⁵⁷ Leah Findlater and Krzysztof Z. Gajos, “Design Space and Evaluation Challenges of Adaptive Graphical User Interfaces,” *AI Magazine* 30, no. 4 (Winter 2009): 68-73.

new, dynamic threats quickly.⁵⁸ The proper development of a UI utilizing a suitable architecture is the key to unlocking the full potential of a functionally adaptive UI.

Architectures play a fundamental role in self-adaptive software systems. Five types of architecture dominate the model-driven UI development architecture references: 3-layer architecture, cameleon-RT, cedar, fame, and malai. 3-layer architecture supports direct adaptations where sensors read information, and targets the environmental context. This architectural reference does not allow for immediate user feedback adaptation but uses manual user input feedback to adapt to future similar situations.⁵⁹ Cameleon-RT is a plasticity architecture that interfaces with interactive spaces. It allows for new adaptive behavior (both direct and indirect) to be added during use. System programmers collect data on the system, user, platform, and environment.⁶⁰ An automated process feeds that data into the architecture as an additional adaptive UI measure. Cedar is a reference architecture for clients interested in developing adaptive enterprise application UIS based on an interpreted software use model-driven approach.⁶¹ Similar to the cameleon-RD architecture, cedar has both direct and indirect adaptation, however the difference is in how the relational database pulls and stores the adaptive behavior information. The fame architecture targets adaptive multimodal UIs using a set of context models in combination

⁵⁸ Scott et al., 2.

⁵⁹ Akiki et al., 9-15.

⁶⁰ Ibid.

⁶¹ Ibid.

with user inputs.⁶² Developers did not design this architecture as a general-purpose reference for adapting other UI characteristics. It supports only procedural adaptation. Fame supports multiple data sources, but relies on user inputs that feed into related models. Lastly, the malai architectural model forms a basis for a technique that uses aspect-oriented modeling (AOM) for adapting UIs.⁶³ Adaptive behavior in the malai architecture is poor because the developer defines the adaptation feedback after the software is used. It relies on modeling code and does not allow for multiple sources for acquiring adaptive behavior data. Although the malai architecture can support direct and indirect adaptations, the user cannot define feedback on the adaptations.

DCGS-A Training Design

Army units conducting AARs indicate training on new DCGS-A software is a challenge due to its enormous capability.⁶⁴ The Army uses a training concept for soldiers to learn how to operate the DCGS-A software. Training concepts are typical for military software, but the DCGS-A software is atypical because it is a system of systems and its training concept is design to train everyone from the seasoned user to users that have never used the software. Intuitive software like Palantir uses a one-time or short course refresher training concept to familiarize the user with basic functions of the software. This type of training concept provides the user with a foundational understanding and allows the user to depend on the intuitive software to build on the foundational skills.

⁶²Ibid.

⁶³ Akiki et al., 9-16.

⁶⁴ Gregor, 13; 34th Infantry Division.

Advanced Palantir training courses provide a train-the-trainer concept where participants take the skills learned back to their units to teach others to use the software. The DCGS-A training concept is one that perceives an untrained user does not have the ability to use the system proficiently without an initial training block and continued reinforced training across all components of the army.⁶⁵ Designers of the training model divide it into seven types across four phases. Each phase segregates the training types and focuses on standardized methods to train individuals. Different agencies within the Army provide training. The type of training determines the agency that will provide, coordinate, and-or assist with training on DCGS-A.

Training is broken down into four phases. Phase one focuses on establishing the skills necessary to operate the DCGS-A software. The training during phase one takes place at institutions, mission-training complexes (MTC), through interactive multimedia instruction (IMI), and through the new equipment training (NET) concept. The NET concept provides the initial transfer of knowledge on the operation and maintenance of DCGS-A equipment from the DCGS-A material developer (MATDEV) to all individuals involved with the training domain. NET trains soldiers on how to operate and maintain DCGS-A equipment.⁶⁶ Phase two integrates the skills previously learned through mission command system integration (MCSI), and collective training. The DCGS-A training concept relies on training to sustain skills whereas intuitive software programs rely on

⁶⁵ United States Army Intelligence Center of Excellence–Military Intelligence School, *Distributed Common Ground System–Army V3, Version 1.0*, November 6, 2013, accessed January 3, 2014, <https://rdl.train.army.mil/catalog-ws/view/100.ATSC/155E7214-A5B8-4870-B16B-4102FE998330-1383762781051/doc.pdf>.

⁶⁶ Ibid.

continued use of the software to retain skills. Sustaining skills is the goal of phase three. It attempts to accomplish this objective through soldier military professional military education (PME), collective training, IMIs, and MTCs. Phase four is referred to as delta training and is facilitated through MTCs, IMI, and NET. Delta training includes NET and relies upon a training support package (TSP).⁶⁷ MATDEV is responsible for the deployment of the TSP and includes training for all DCGS-A configurations, in all training domains. The TSP trains and sustains individual collective tasks and skills to develop proficient soldiers, leaders, staffs, and units.

In addition to the four phases of the training are the institution, operational, and self-development domains. The institutional domain concentrates on training operator and maintainer-level individual tasks. It exposes soldiers to mission-oriented collective tasks training and teaches leaders to leader task training. The operational domain focuses on system and collective-level tasks utilizing the DCGS-A IMIs, training aids, devices, simulators, and simulations (TADSS). The self-development domain offers training on individual tasks through IMIs, TADSS, DCGS-A embedded TSP and soldier training package (STP), and the DCGS-A TM.⁶⁸ The self-development domain is broken down into structured, guided, and personal self-development. Structured self-development consists of the training one receives throughout their career in the military and parallels closely to classroom and on-the-job learning. Guided self-development is optional learning that will help keep individuals prepared for changing technical, functional, and

⁶⁷ Ibid.

⁶⁸ Ibid.

leadership responsibilities throughout their career. Personal self-development is self-initiated learning where the individual defines the objective, pace, and process. The embedded DCGS- A TSP enables the self-development strategy.⁶⁹

The Army provides DCGS-A training at various sites through different methods. The foundry program is an army G2/INSCOM led resourced training program that assists commanders and G2s by serving as the foundation and coordination point to obtain military intelligence training opportunities not available at the tactical echelon or through the institutional training system. Foundry provides an interim solution to military intelligence systems and concepts to assist commanders with sustainment training opportunities prior to objective DCGS-A TSP development and delivery. Mission training complexes provide battle command and staff training, training support, and publications to soldiers and units to prepare for unified land operations (ULO) in joint-interagency-intergovernmental-multinational (JIIM) operations. Mission Command Training Program (MCTP) conducts or supports combined arms training that replicate JIIM in a contemporary operating environment (COE), at worldwide locations, in accordance with the ARFORGEN model. Training is provided by brigade combat teams (BCTs), divisions, corps, army service component commands (ASCCs), joint force land component commander (JFLCCs), and joint task forces (JTFs) in order to create training experiences that enable the Army's senior mission commanders to develop current, relevant, campaign-quality, joint and expeditionary mission command instincts and skills. Combat training centers (CTCs) enable units to train in ULO, executed through decisive

⁶⁹ Ibid.

action by performing army core competencies, guided by mission command. Units will further their DCGS-A skills at CTCs by utilizing CTC-fielded DCGS-A systems during their exercises.

The Army National Guard (ARNG) and the U.S. Army Reserves (USAR) receive component-specific training on DCGS-A. Soldier's DCGS-A skills are more perishable in ARNG and USAR due to the frequency of their training. ARNG and USAR soldiers minimally meet one weekend each month, and two weeks per year. Optimally, these soldiers should be able to train on DCGS-A each month, however mission requirements and access to DCGS-A training frequently makes it more challenging to sustain their skills. To make DCGS-A training more available to ARNG elements, units have the ability utilize regional training institutes at Camp Williams, Utah, and Camp Clay, Georgia in addition to the institutional domain. These training sites maintain operational DCGS-A software platforms for individuals to receive DCGS-A training. Fort Devens, Massachusetts offers those affiliated with USAR an opportunity to train on DCGS-A by USAR The Army School System (TASS) Battalions. There are regional locations known as Army Reserve Intelligence Support Centers (ARISC) throughout the United States that have sites available for USAR soldiers to train on DCGS-A as well.

The United States Army Engineer School (USAES) plans to create a TRADOC capabilities manager (TCM) geospatial lab. This lab will field the most recent version of the geographic intelligence (GEOINT) workstation (GWS) software and hardware suites. The goal of the lab is to analyze and assess geospatial engineer problem sets and produce tactics, techniques, and procedures to reduce hindrance of newly fielded DCGS-A software and hardware solutions.

To ensure the achievement of institutional domain training, the army fields DCGS-A training systems at various institutions. As of 2013, DCGS-A was fielded at the following: (1) Space and missile defense/army forces strategic command; (2) US Army sergeant major academy; (3) TRADOC centers of excellence; (4) Warrant officer career college; (5) Command and General Staff College—digital learning development center; (6) JFC special warfare center and school; (7) C2 warrior school/Army joint support team; (8) The 505th air training wing at Nellis Air Force base (9) The 2nd combat weather systems squadron; and (10) The Air Force reserve center. In total, the Army fields 115 systems across a variety of unique training facilities.

The Army trains enlisted, warrant officer, officer, and Air Force officers with specific MOSs on DCGS-A software. Table 1 provides a list, by military occupation specialty (MOS), of the soldiers and airmen that receive training at their respective training course.

Summary

This chapter provided a review of the common software failure characteristics, the insights into case studies that depicted software achievements and failures, insights into modern interface design models, and an overview of the DCGS-A enterprise training model. Experts in the field of information technology and software design agree the most common characteristics of software failure identified that human error is the most common reason why software programs fail. Avoiding these areas of failure is the responsibility of program managers. The case studies gave insight into the real-world application of the software failure characteristics, and provided a contextual background of software achievements. Introducing the software development models provided a

comprehensive understanding into how software developers design modern software framework and the role software architecture plays into developing adaptive software programs. The expert testimony imparts great knowledge into developing adaptive software that users will find easy to use and intuitive. Program managers can use this knowledge to apply it to future software development, when planning for the development of easy to use, and adaptive (or intuitive) software. A review of the DCGS-A training program identified the extensive training opportunities and methods of skill retention available to DCGS-A users.

The next chapter will present the research methodology used in this study by the researcher to collect data. The researcher will introduce the research model, the method for collecting the data, identification of the population and sample size, and the method for data analysis.

CHAPTER 3

RESEARCH METHODOLOGY

Overview

This study focuses on issues that affect training and using the Distributed Common Ground System-Army (DCGS-A) system and the research attempts to determine if an improved user interface (UI) will increase user efficiency. This chapter provides reasons for the use of a survey methodology to collect data on this topic. The chapter will provide insight into the survey instrumentation, data collection and analysis methodologies, and the selected population.

The primary research question was, “How does the UI within DCGS-A affect operability of the system?” In order to answer this question, the researcher solicited data from intelligence branched officers attending the resident Command and General Staff College (CGSC) at Fort Leavenworth, Kansas. The researcher solicited data to answer the following secondary questions:

1. What other issues affect the operability of DCGS-A?
2. Does the UI of the current version of DCGS-A detract from the capability to analyze and disseminate intelligence in a timely fashion?
 - a. If yes, what part(s) of the UI hinder the ability?
 - b. What version of DCGS-A is being used/trained on?

Is the DCGS-A UI easy to use?

- c. How could the DCGS-A UI be improved?
- d. What would make the DCGS-A UI more functional?

3. Does the amount of time spent using and-or training on DCGS-A hinder the user's ability to operate the system?
- a. How much time was spent using DCGS-A other than for training?
 - b. How much time was spent on DCGS-A over the past year?
 - c. Is the amount of DCGS-A training received sufficient to provide the skills to operate the system?
 - d. Can a change in the UI allow easier user functionality with less training time?

Survey Design

This study used the survey research model as described by Floyd J. Fowler Jr. (2009) in *Survey Research Methods*, and Arlene Fink (2009) in *How to Conduct Surveys: A Step-by-Step Guide*. "Surveys are designed to produce statistics about a target population."⁷⁰ The cross-sectional survey used in this study is a survey designed to collect data at a sign point in time.

It seeks to collect data from the population to answer if the current DCGS-A UI is easy to use. Data from a cross-sectional survey will describe issues as they are in the current frame of time. This survey is attempting to understand the complexities behind the current issues that plague the DCGS-A system. Results from a cross-sectional survey are short lived when changes are rapid. Such is the case with DCGS-A because the system is constantly being updated with patches and upgrades.

⁷⁰ Floyd J. Fowler Jr., *Survey Research Methods* (Thousand Oaks, CA: Sage Publications, 2009), 1.

Online Survey

Online surveys are normally self-administered and are easy manage because of access and availability. They have the ability to reach large audiences throughout the world and provide time for thoughtful answers. Data from online surveys are quickly compiled and easy to process when using certified and reliable survey software.⁷¹ Populations receive an email that contains a survey link that, when accessed, provides the participant access to the online survey. The researcher acquired the population's email addresses from the CGSC's registrar and entered into the survey software by the CGSC Quality Control Office. The participant can access the survey from any computer that has access to the internet. Accessing the online survey is anonymous and ensures confidentiality.

Survey Instrument

The online survey "Distributed Common Ground System–Army (2015)" had three areas of concentration: (1) The amount of time spent training on DCGS-A; (2) The amount of time using DCGS-A other than for training; and (3) DCGS-A workspace and UI. The online survey consisted of several closed-ended questions in the form of multiple-choice answers, Likert scale ratings, dichotomous questions (yes/no), and demographic questions. "Closed-ended questions provide respondents with a list of response options from which to choose."⁷² The online survey also included several open-

⁷¹ Fowler, 83.

⁷² Valarie M. Sue and Lois A. Ritter, *Conducting Online Surveys* (Thousand Oaks, CA: Sage Publications, 2012), 60.

ended questions where participants were able to expound on their closed-ended question responses. “These questions allow participants to answer in their own words by typing in their response in an empty box.”⁷³

Quantitative

The researcher designed the closed-ended questions to confirm or deny assumptions the research question poses. A qualifying contingency question asked if the participant received training on the DCGS-A system. A “No” response would disqualify the participant from the survey because the researcher seeks data from participants that have experience using DCGS-A. Demographic questions ask for the participant’s service component (regular Army, National Guard, or Army Reserves), and military occupation specialty (MOS).

A dichotomous question asked if the participant had experience using DCGS-A during times other than training. Likert scale questions ask how long the participant has been using DCGS-A and how much time was spent using DCGS-A during times other than training over the past year. The goal of these Likert scale questions is to determine the length of experience the participant has using DCGS-A. Another dichotomous question asked the participant if the amount of DCGS-A training they receive is sufficient to provide them with the skills they need to operate the system. Another Likert scale question asked how much time has the participant spent training on DCGS-A. These questions aim to understand if DCGS-A training is enough to retain the skills for system operation. The next set of dichotomous questions asked about DCGS-A use while

⁷³ Sue and Ritter, 56.

deployed. A correlating Likert scale question asked how much time was spent using DCGS-A during the participant's most recent deployment. The research was attempting to understand the relationship of DCGS-A use while deployed to the ease-of-use on DCGS-A. Another set of dichotomous questions ask if an integrated DCGS-A workspace would make DCGS-A easier to use or more functional. The questions sought to determine the relationship between the current version of DCGS-A and if the experience of the participant directly relates to a change in the workspace design of DCGS-A. Another set of dichotomous questions ask if bandwidth problems affect the user functionality, or if hardware problems affect user functionality on DCGS-A. The research attempted to understand if there are other issues besides what the research question is asking that affects DCGS-A ease-of-use. A demographic question asked which version of DCGS-A the participant used or trained on, and provides an option if they are unsure. A final dichotomous question asked if the DCGS-A UI is easy to use. This is a culminating question that the researcher posed in attempt to extract a summary of the participant's feelings towards the DCGS-A UI. Lastly, the research asked the participant to answer fifty-four Likert scale questions to identify if any of the DCGS-A workspace tools and widgets need substantial improvement, some improvement, or no improvement. The research provided an option to the participant if they have no experience with the tool or widget.

Qualitative

The online survey provided open-ended question comment boxes to allow participants the option to provide additional feedback to elaborate on their closed-ended question responses. The researcher provided open-ended question in the following

sections of the online survey: (1) The participant's experience with training on DCGS-A (page 4); (2) The participant's use of DCGS-A while deployed (page 5); (3) DCGS-A workspace, ease of use and functionality (pages 6 and 7); and (4) DCGS-A UI improvement (page 9).

Demographics

The Inquisite Survey Builder allowed for building the survey questions in a variety of ways. The researcher was able to manipulate the software to design the survey questions and the online survey itself. Demographics were included so the researcher could later group and file the participants by their responses. The demographic questions included: (1) Service component (regular Army, National Guard, Army Reserves); (2) Rank type (officer, enlisted, warrant officer); and (3) Military Occupation Specialty (MOS). A question later in the survey asks if the participant has recent deployment experience.

Population and Sampling

The researcher conducted this study at the United States Army Command and General Staff College. CGSC is a graduate level school accredited by the North Central Association (NCA) to provide college credit hours and grant master's degrees. The population selected to participate in this study was a group of military intelligence officers attending the Command and General Staff Officers' Course (CGSOC) of the Command and General Staff School located at Fort Leavenworth, Kansas.

“A population is the entire assembly of individuals, groups, or objects that generalize results.”⁷⁴ Identification and generation of the population the researcher selected for the survey will produce the sample frame. The sample is the individuals selected from the sample frame to participate in the survey. The CGSC Quality Assurance Office input the sample frame into the Inquisite Survey Builder and the software randomly selects the sample.

This online survey used probability sampling as a method for classifying the sampling. Probability sampling is the random selection of participants from a defined sampling frame, and affords the researcher the opportunity to reach conclusions about population characteristics bas on sample statistics.⁷⁵ The population contained primarily United States Army military officers that were mostly Majors (a field grade officer rank), but may have contained a small number of Captains and-or Lieutenant Colonels. The students attending the resident CGSOC have an average of 32-43 years of age, have at least a Bachelor’s degree, and on average have between ten to fourteen years of service in the military. The classrooms are broke down into groups of sixteen students where each classroom has at least one female, a different service component (Navy, Marines, and Air Force) officer, an international military officer, and an officer from either the National Guard or Army Reserves. Some classes have a civilian from an intergovernmental organization that has the government pay scale of at least GS-13. Most of the military

⁷⁴ Sue and Ritter, 33.

⁷⁵ Ibid., 36.

students have served at least one combat tour, while many have completed multiple combat tours.

Starting in academic year (AY) 2015, CGSC offered one start date for attending CGSOC. CGSC identifies this class as class 2015. The students arrived during July of calendar year 2014 and consisted of approximately 1,100 students. The students attending class 14-02 graduated in December 2014 and the researcher did not consider this class as part of the sample frame. The researcher administered the final survey to seventy-four of the AY 2015 students (United States military only), that were branched as military intelligence officer. The research did not include international officers, intergovernmental civilians, or other service component officers in the administration of the final survey. This population was accessible by email invitations.

Margin of Error and Confidence Level

The margin of error is the estimates from the data are correct to within a small margin of variability.⁷⁶ The researcher has a moderate level of confidence the population will represent the overall officer population within the United States Army. The population selected for the research contained seventy-four military intelligence officers attending CGSOC at Fort Leavenworth, Kansas. Using a response distribution of 50 percent, the researcher desired sixty-three participants to respond in order to achieve a 95 percent confidence interval with a ± 5 percent margin of error.

⁷⁶ Ibid., 41.

Survey Administration

The CGSC Quality Assurance Office provided access and use of a survey program that allowed the researcher to build the survey and administer it to the population. The researcher used the Inquisite Survey Builder, which is a licensed software for designing surveys. This software is developed and owned by the Allegiance Software company. The Allegiance Engage platform is a system for collecting feedback and analysis on the survey data. This platform publishes and administers the online survey. The platform secured all data to meet human subjects' protections and Army regulatory requirements for collecting data from active duty members. It allows the CGSC Quality Assurance Office to upload an email roster from the sample frame. The software converts the emails into special codes and the software uses the codes to identify those that did not complete the survey. Emails and names of invitees did not appear in any data downloaded from the software. The software removes any personal identifiers from data reports, thus providing a high level of confidentiality.

An informed consent notifies participants on the general nature of the survey, the identity of the researcher, how the researcher is going to use the data, the average length of time to complete the survey, and if there is any risk involved in participating in the survey.⁷⁷ The email invitation and the opening page on the survey provided the informed consent to the participant. The researcher provided contact information for the CGSC Quality Assurance Office, should the participant have questions concerning participation

⁷⁷ Sue and Ritter, 28.

in the study. The Allegiance Engage platform delivered an email with a web link to the survey to the invitees.

Protection of Human Rights

The researcher completed an application to conduct research with oversight from the CGSC Quality Assurance Office. The CGSC Human Protection Administrator and the Institutional Review Board (IRB) determined this research to be exempt from the human subject review board. The researcher did not handle any of the population's personally identifying information (PII). The CGSC registrar's office sent the population's email addresses directly to the CGSC Quality Assurance Office. The CGSC Quality Assurance Office input the population's email addresses directly into the Inquisite survey software. Results and data from the survey did not include any PII or email addresses. This research sought to compile data and did not seek to identify who the participant and nonparticipants were.

Data Analysis

Quantitative

The quantitative data will be analyzed using descriptive statistics. Descriptive statistics provide simple summaries about the sample and the responses to the questions.⁷⁸ Frequencies, measures of central tendency (the mean, median, and mode), and measures of variation (range and standard deviation), along with graphics analysis will outline the results of the collected data.

⁷⁸ Arlene Fink, *How to Conduct Surveys: A Step-by-Step Guide* (Thousand Oaks, CA: Sage Publications, 2009), 78.

Qualitative

The qualitative data will be analyzed using content analysis. The analysis of the open-ended responses will assist the researcher in categorizing the data from the respondents. The researcher's committee provided additional oversight during data analysis to minimize the researcher's interpretation bias. The open-ended questions in the survey facilitate the collection of qualitative data that revolved around two broad categories: training and UI improvement. The researcher designed open-ended questions around these categories in order to acquire relevant data for the appropriate theme. After the researcher collected the aggregate data, the researcher placed the data into subcategories and selectively coded following the techniques outlined in *Qualitative Research Practice* by Clive Seale, et al (2007). Data analysis ended when core categories emerged and the researcher was able to integrate the analysis into a developed 'story' that captured the main themes of this study. The researcher designed the open-ended questions to ensure achievement of theoretical saturation in each category. Theoretical saturation takes place when additional data no longer prompts new distinctions or refinements to the emerging theory.⁷⁹ The researcher will code the data using the selective coding method. The selective coding method involves selecting the core category and systematically relating it to the other categories then filling in those categories that need further refinement and development.⁸⁰ Coding the data this way will

⁷⁹ Ian Dey, "Grounded Theory," in *Qualitative Research Practice*, ed. Clive Seale et al. (Thousand Oaks, CA: Sage Publications, 2007), 81.

⁸⁰ Ibid.

formulate the data into categories that can be analyzed where results will be published in chapter 4 of this study.

Summary

This chapter presented the methodology used by the researcher to collect data for this study. The survey instrument is key facilitating the questions from the researcher, but the results are perishable because the Army continues to improve and upgrade the DCGS-A system. The researcher chose to collect both quantitative and qualitative data in attempt to substantiate the results and provide meaning behind the quantitative data. As long as the anticipated population sample responds to the survey, the researcher will maintain a moderate level of confidence the data will represent the majority of U.S. Army officers. After the survey has closed, the researcher will collect the data, analyze it and then group it into qualitative categories. The researcher will measure the quantitative data through statistical analysis. The next chapter will present the results of the survey data analysis.

CHAPTER 4

SURVEY RESULTS

Overview

This study examined the UI of the Distributed Common Ground System – Army (DCGS-A) system, its training program, and the ease of the user’s ability to retain their skills. It specifically identified the common characteristics found in failed software programs, mostly which are due to human factors. The case studies support the finding that software failures occur due to a combination of human factors, but software achievements are successful due to human decisions. The research explored software designs, identifying that using a model-based user interface development (MBUID) design is the best approach to address specialized solutions for effectively capturing UI functionality. It detailed the features and designs of recent developments in the intuitive UI field that identifies several graphical user interface (GUI) models that designers can use when crafting user interfaces to improve functionality and ease-of-use. The research also acknowledged the Army’s current DCGS-A training program that encompasses all the areas of training from initial user training to user sustainment training. The researcher used a self-administered online survey technique to collect the information needed for this research. The survey asked students attending the Command and General Staff Officers’ College (CGSOC) if the DCGS-A UI is intuitive, easy to use, and functional, the amount of time spent using system in both garrison and deployed environments, and the amount of time spent training on the system. The survey also asked the respondents if the tools within the system needed to be improved. This chapter provides the survey results.

Research Questions

The primary research question was, “How does the UI within DCGS-A affect operability of the system?” To answer this question, the researcher solicited data from students attending CGSOC on the following secondary questions:

1. What other issues affect the operability of DCGS-A?
2. Does the UI of the current version of DCGS-A detract from the capability to analyze and disseminate intelligence in a timely fashion?
 - a. If yes, what part(s) of the UI hinder the ability?
 - b. What version of DCGS-A is being used/trained on?

Is the DCGS-A UI easy to use?
 - c. How could the DCGS-A UI be improved?
 - d. What would make the DCGS-A UI more functional?
3. Does the amount of time spent using and-or training on DCGS-A hinder the user’s ability to operate the system?
 - a. How much time was spent using DCGS-A other than for training?
 - b. How much time was spent on DCGS-A over the past year?
 - c. Is the amount of DCGS-A training received sufficient to provide the skills to operate the system?
 - d. Can a change in the UI allow easier user functionality with less training time?

Survey

The survey, “DCGS-A Use and Training” had three areas of concentration:

- (1) The amount of time spent training on DCGS-A; (2) The amount of time using DCGS-

A other than for training; and (3) DCGS-A workspace and UI. The online survey consisted of several closed-ended questions in the form of multiple-choice answers, Likert scale ratings, dichotomous questions (yes/no), and demographic questions.

The online survey also included several open-ended questions where participants were able to expound on their closed-ended question responses. The survey provided nine open-ended opportunities to the participants:

1. Comments on DCGS-A training? This question follows the dichotomous question, “Is the amount of DCGS-A training you receive sufficient to provide you with the skills to operate the system?”, and the Likert scale question, “How much time have you spent training on DCGS-A over the past year?”
2. How is your ability to operate DCGS-A affected by the amount of time between uses. This question follows the Likert scale question, “How much time did you spend using DCGS-A on your most recent deployment?”
3. What combined components of DCGS-A would make DCGS-A easier to use? Following a provided definition of what an integrated DCGS-A workspace is, this question follows the dichotomous question, “Would an integrated DCGS-A workspace make DCGS-A easier to use?”
4. What combined components of DCGS-A would make DCGS-A more functional? Following a provided definition of what an integrated DCGS-A workspace is, this question follows the dichotomous question, “Would an integrated DCGS-A workspace make DCGS-A more functional?”
5. How does bandwidth affect functionality of DCGS-A? This question follows the dichotomous question, “Do bandwidth problems affect user functionality of

DCGS-A?” This question appears on the online survey only if the participant answers “Yes” to the dichotomous question.

6. How does hardware problems affect the functionality of DCGS-A? This question follows the dichotomous question, “Do hardware (i.e. servers, computers, etc...) problems affect user functionality of DCGS-A?” This question appears on the online survey only if the participant answers “Yes” to the dichotomous question.
7. How could the DCGS-A UI be improved? This question follows the dichotomous question, “Is the DCGS-A user interface easy to use?”
8. What would make the DCGS-A user interface more functional? This question directly follows the previous open-ended question, “How could the DCGS-A user interface be improved?”
9. Comments. The comments question is the last question in the online survey and provides the participant a final opportunity to express any thoughts that the survey did not capture.

Survey Participants

From the population of seventy-four military intelligence officers attending the academic year (AY) 2015 Command and General Staff College (CSGS), nineteen participants responded. Of those nineteen participants, fourteen qualified to complete the survey. The fourteen participants created a low confidence interval of ± 8.94 , meaning the researcher has 95 percent confidence that the true population proportion falls within the 10 to 28 percent range.

The researcher designed the first question within the online survey to be a qualifying question. A qualifying question is one that asks the participant if they have a particular type of experience or skill that relates to the survey. If the participant has the experience or skills that the online survey asks in the qualifying question and they answer the qualifying question truthfully, the participant will be able to answer the remaining questions in the survey. Conversely, if the participant does not have the experience or skills the researcher is seeking and the participant answers the qualifying question truthfully, then the survey will send the participant to a “Thank you for participating” screen, not allowing the participant to answer any further survey questions. Nineteen of the seventy-four participants that responded, fourteen answered “Yes” to the qualifying question.

The fourteen respondents that qualified to continue the survey were commissioned officers serving in the regular army. Twelve of the fourteen (86 percent) held the military occupation specialty (MOS) of 35D–All-Source Intelligence officer, one (7 percent) was a 35E–Counterintelligence/Human Intelligence officer, and one (7 percent) was a 35G–Signals Intelligence/Electronic Warfare officer. Table 1 depicts the qualification questions results by count and percent. Table 2 depicts the population demographics.

Table 1. Qualifying Question			
Question	Metric	Yes	No
Have you been trained to use the Distributed Common Ground System-Army (DCGS-A) system?	Count	14	5
	Percent	74	26

Source: Created by author.

Table 2. Population Demographics				
Question	Metric	35D-All-Source Intelligence Officer	35E-CI/HUMINT Officer	35G-Signals Intelligence/Electronic Warfare Officer
What is your military occupation specialty?	Count	12	1	1
	Percent	86	7	7

Source: Created by author

Eight (57 percent) of the participants responded they had experience using DCGS-A during times other than training. Three (21 percent) did not answer the question, and the other three (21 percent) said their only experience using DCGS-A was during training. Table 3 depicts the results of the participants' experience using DCGS-A during times other than training. Four (29 percent) of the fourteen participants said they had spent more than five years using DCGS-A, while two (14 percent) of the fourteen participants indicated they have spent at least three but not more than five years using DCGS-A. Three (21 percent) of the fourteen indicated they have spent at least one but less than three years using DCGS-A, and the last five (36 percent) of the fourteen participants said they had spent less than a year using the system. Table 4 depicts the participants' level of experience using DCGS-A. Of the eight (57 percent) participants who said they used DCGS-A during their most recent deployment, five (36 percent) said

they used the system for at least six but not more than twelve months while the other three (14 percent) said they used the system for up to two months. Table 5 shows the amount of participant experience using DCGS-A while deployed.

Table 3. Using DCGS-A During Times Other Than For Training				
Question	Metric	Yes	No	Did not Answer
Do you have experience using DCGS-A during times other than training?	Count	8	3	3
	Percent	57	21	21

Source: Created by author.

Table 4. Experience Using DCGS-A					
Question	Metric	Less than 1 year	At least 1 but less than 3 years	At least 3 but less than 5 years	More than 5 years
How long have you been using DCGS-A?	Count	5	3	2	4
	Percent	36	21	14	29

Source: Created by author.

Table 5. Experience Using DCGS-A While Deployed					
Question	Metric	None	Up to two months	At least 2 but not more than 6 months	At least 6 but not more than 12 months
How much time have you spent using DCGS-A on your most recent deployment?	Count	6	3	0	5
	Percent	43	21	0	36

Source: Created by author.

Research Question 1: What other issues affect
the operability of DCGS-A?

To identify issues that affect the functionality and operability of DCGS-A, the survey asked participants to respond to two dichotomous questions and provided open-ended questions so they could expound on their responses. Ten (71 percent) of the participants indicated bandwidth problems affected the functionality of DCGS-A.

Likewise, ten (71 percent) of the participants indicated that hardware problems affected the functionality of DCGS-A. Table 6 depicts the count and percent responses from the participants.

Table 6. Other Issues Affecting DCGS-A Operability			
Question	Count	Yes	No
Do bandwidth problems affect user functionality of DCGS-A?	Count	10	4
	Percent	71	29
Do hardware (i.e. servers, computers, etc...) problems affect user functionality of DCGS-A?	Count	10	4
	Percent	71	29

Source: Created by author.

Open-Ended Question: How does bandwidth
affect functionality of DCGS-A?

Ten (71 percent) participants responded to the open-ended question with the theme that imagery tends to tie up the bandwidth that is dedicated for DCGS-A, and there is normally a limited amount of bandwidth to use the system to its full capacity. Two of the participant's comments capture the essence of the bandwidth problems:

We tried to set [DCGS-A] up at a BN MAIN CP in Germany, and it worked extremely slow, or would drop off the network due to limited bandwidth available.

Maps do not load properly or take extremely long amounts of time to load into the computer. Internet access is slow. [The] Trojan Spirit cannot support large numbers of DCGS-A computer on its system and bandwidth is low.

Open-Ended Question: How does hardware problems affect the functionality of DCGS-A?

Seven (50 percent) of the participants provided comments that indicated the components of DCGS-A needs better servicing and maintenance at the unit level. The premise of the discussions suggests the hardware components of DCGS-A do not measure up to the rigors of mission requirements. One participant provided the following account:

ACT-E hardware does not work or is [not] properly serviced at the tactical level. FSRs do not understand how to maintain the equipment in the ACT-E, they only know about maintenance of the server stack.

Research Question 2: Does the user interface of the current version of DCGS-A detract from the capability to analyze and disseminate intelligence in a timely fashion?

DCGS-A Tools

The researcher sought to answer this question by providing participants an opportunity to identify any part of the DCGS-A UI that makes it difficult to use. Using a Likert scale rating, the survey asked participants if the following categories of tools within DCGS-A needed no improvement or needed some level of improvement:

1. Geospatial tools: These tools included 3d Fly Thru, ArGIS10, Geospatial Analysis, and Map services and product dissemination.
- 2, Collection management: The tools that were included in this category were ISR sync matrix, Mission planning/management, Requirements management, RFI, Roles, users, units, and privileges setup, and User management.

3. Full motion video: The tools included in this category included MIScreener, MISnapReporter, and Socet GXP.
4. HUMINT: The tools included in this category were Break down PIRs, Export to HOT-R, HUMINT collection requirement (sphere of influence) source visualization, Unit task order, and Unit status.
5. Ozone widgets: The Ozone widgets included the following tools: ATHEN-A report widget, ATHEN-A message widget, ATHEN-A status widget, BAT dossier widget, BAT search widget, Cascading query widget, Common map widget, DIB query widget, DIMS-F dossier widget, DIMS-F search widget, GeoDiscover, Geo Package, GeoPrint, GeoWebPrint, GeoWebView, IWEDA widget, Weather overlays widget, and Weather report widget.
6. All-source: The tools included in the all-source category included 2D map, AgileGraphic tools, Analyst notebook import, Automated product tool (APT), Excel importer, Fusion exploitation framework (FEF), Create NAI, Geospacial query w/in Linear XOI, Link analysis tool, Map setup, MFWS startup time, Remarks viewer, Save DB changes UI/workflow, Search help file, Text extraction, User configuration templates, and Visual analyst tools.

Statistical Analysis

The researcher collected the responses from the Likert scale DCGS-A tools section and used a likelihood ratio test to analyze the responses. The Kruskal-Wallis test provided no significant statistical results when comparing the data from the DCGS-A tools to the dichotomous and multiple-choice questions due to the low participation rate. The Mann-Whitney test provided no valuable statistical results also due to the low

participation and response rate. In order to understand the results, the researcher applied a likelihood ratio test against the responses the participants provided with alpha p-tests of both .05 and .10. The researcher compared the p-values against the actual alpha values, which produced statistically significant results in some cases. When conducting a p-test against the alpha value, and the p-test is less than the alpha value ($\alpha < .05$, or $\alpha < .10$), then the null hypothesis is rejected. The null hypothesis in this case is that the tool does not need improvement. The .05 alpha p-test provides more reliable data than the .10 alpha p-test. The .10 alpha p-test provided a handful more results than the results from the .05 alpha p-test.

Table 7 depicts the statistical results of the DCGS-A tools using a likelihood ratio test. The researcher combined the “Needs some improvement” and “Needs substantial improvement” Likert scale ratings to simplify the results from the DCGS-A tools section of the survey

Table 7. DCGS-A Tools							
Category	Question	Needs Improvement	Responses	Likelihood Ratio	P-Test $\alpha = .05$	P-Test $\alpha = .10$	Alpha
Geospatial Tools	ArcGIS 10	10	13	7.3028	1	1	0.0461426
Geospatial Tools	Map services and product dissemination	9	12	4.8054	0	1	0.0729980
Geospatial Tools	Geospacial Analysis	8	10	6.8719	0	1	0.0546875
Geospatial Tools	3d Fly Thru	4	9	1.0572	0	0	0.7460938
Collection Mgt	Mission planning/management	8	10	6.8719	0	1	0.0546875
Collection Mgt	User management	7	9	4.3535	0	1	0.0898438
Collection Mgt	Requirements management	6	8	2.8477	0	0	0.1445313
Collection Mgt	Roles, users, units, and privileges setup	7	8	12.5663	1	1	0.0351563
Collection Mgt	ISR sync matrix	5	8	1.2875	0	0	0.3632813
Collection Mgt	RFI	5	8	1.2875	0	0	0.3632813
FMV	Socet GXP	3	6	1.0000	0	0	0.6562500

FMV	MIScreener	2	5	1.1059	0	0	0.8125000
FMV	MISnapReporter	3	4	1.6875	0	0	0.3125000
HUMINT	Break down PIRs	5	6	4.2867	0	0	0.1093750
HUMINT	Export to HOT-R	3	6	1.0000	0	0	0.6562500
HUMINT	Unit task order	4	6	1.4047	0	0	0.3437500
HUMINT	HUMINT collection requirement (sphere of influence) source visualization	3	5	1.1059	0	0	0.5000000
HUMINT	Unit status	2	5	1.1059	0	0	0.8125000
Ozone Widgets	DIB query widget	2	7	1.9428	0	0	0.9375000
Ozone Widgets	GeoPackage	2	6	1.4047	0	0	0.8906250
Ozone Widgets	Weather report widget	3	6	1.0000	0	0	0.6562500
Ozone Widgets	Common map widget	3	5	1.1059	0	0	0.5000000
Ozone Widgets	GeoDiscover	3	5	1.1059	0	0	0.5000000
Ozone Widgets	GeoPrint	2	5	1.1059	0	0	0.8125000
Ozone Widgets	GeoWebPrint	3	5	1.1059	0	0	0.5000000
Ozone Widgets	GeoWebView	2	5	1.1059	0	0	0.8125000
Ozone Widgets	Weather overlays widget	3	5	1.1059	0	0	0.5000000
Ozone Widgets	BAT dossier widget	3	4	1.6875	0	0	0.3125000
Ozone Widgets	BAT search widget	3	4	1.6875	0	0	0.3125000
Ozone Widgets	Cascading query widget	2	4	1.0000	0	0	0.6875000
Ozone Widgets	ATHEN-A report widget	2	3	1.1852	0	0	0.5000000
Ozone Widgets	ATHEN-A message widget	1	3	1.1852	0	0	0.8750000
Ozone Widgets	ATHEN-A status widget	1	3	1.1852	0	0	0.8750000
Ozone Widgets	DIMS-F dossier widget	2	3	1.1852	0	0	0.6875000
Ozone Widgets	DIMS-F search widget	2	3	1.1852	0	0	0.6875000
Ozone Widgets	IWEDA widget	1	3	1.1852	0	0	0.8750000
All-Source	2D map	8	11	3.2516	0	0	0.1132813
All-Source	MFWS startup time	7	9	4.3535	0	1	0.0898438
All-Source	Link analysis tool	7	11	1.5133	0	0	0.2744141
All-Source	Map setup	7	11	1.5133	0	0	0.2744141
All-Source	Analyst notebook import	7	10	2.2769	0	0	0.1718750
All-Source	Excel importer	5	10	1.0000	0	0	0.6230469
All-Source	Create NAI	5	10	1.0000	0	0	0.6230469
All-Source	AgileGraphic tools	5	8	1.2875	0	0	0.3632813
All-Source	Remarks viewer	5	8	1.2875	0	0	0.3632813
All-Source	Visual analyst tools	5	8	1.2875	0	0	0.3632813
All-Source	Save DB changes UI/workflow	4	7	1.0743	0	0	0.5000000
All-Source	Search help file	5	7	1.9428	0	0	0.2265625
All-Source	User configuration templates	5	7	1.9428	0	0	0.2265625
All-Source	Automated product tool (APT)	3	6	1.0000	0	0	0.6562500
All-Source	Fusion exploitation framework (FEF)	4	6	1.4047	0	0	0.6562500

All-Source	Geospatial query w/in Linear XOI	4	6	1.4047	0	0	0.6562500
All-Source	Linear XOI	4	6	1.4047	0	0	0.6562500
All-Source	Text extraction	4	6	1.4047	0	0	0.6562500

Source: Created by author.

Workspace Integration

The researcher sought to answer the second part of this research question by asking the participants a series of questions related to the integration of DCGS-A workspace. The survey asked the dichotomous question, “Would an integrated DCGS-A workspace make DCGS-A easier to use?” Nine (64 percent) of the participants answered “Yes” to the question, while the other five (36 percent) answered “No”. This question attempted to determine a popular opinion if by integrating portions of the DCGS-A workspace would make the system easier to use. Table 8 depicts the participants’ count and percent responses to the question.

Table 8. Ease of Use with an Integrated DCGS-A Workspace			
Question	Metric	Yes	No
Would an integrated DCGS-A workspace make DCGS-A easier to use?	Count	9	5
	Percent	64	36

Source: Created by author.

The researcher asked the participants the dichotomous question, “Would an integrated DCGS-A workspace make DCGS-A more functional?” Again, nine (64 percent) of the participants answered “Yes”, while five (36 percent) answered “No”. This question sought to determine if the participants believed integrating portions of the

DCGS-A workspace would make the system more functional. Functionality differs from ease-of-use, whereas functionality is the qualitative purpose the system has, and ease-of-use is how easy an operator can use the system. Table 9 depicts the participants' count and percent responses.

Table 9. A More Functional DCGS-A Workspace			
Question	Metric	Yes	No
Would an integrated DCGS-A workspace make DCGS-A more functional?	Count	9	5
	Percent	64	36

Source: Created by author

Open-Ended Question: What combined components of DCGS-A would make DCGS-A easier to use?

The researcher asked this open-ended question to the participants to provide them with an opportunity to give their opinion on DCGS-A ease-of-use. The resulting information from the eight (57 percent) participants that responded to the question indicated a variety of categorical responses. The first category of the responses indicated a need for DCGS-A to be integrated with other systems.

A second category that emerged from the responses indicated the DCGS-A UI is difficult to use, and a redesign interface comparable to commercially produced software may make it easier to use. This participant's response captures the essence of this category's responses:

[DCGS-A] is still a difficult program to understand requiring lots of experience and training. The only way it would be easier to use would be if it was comparable with Microsoft Office tools and Google Earth.

Open-Ended Question: What combined components of DCGS-A would make DCGS-A more functional?

This open-ended question provided participants with an opportunity to give their opinion on the functionality of DCGS-A. The information collected from the seven (50 percent) participants that responded to the question concluded that DCGS-A should be able to communicate and work with other systems. One participant exclaimed:

CPOF and DCGS-A integration will support functionality. The components would include CPOF, CHARCs-ITRT, CIDNE, NSA-NET, TIGR, BATS, and SATCOM. I included SATCOM because most of the time tactical MI teams are not operating within the FOB. They are sent to the Ops, patrols, and strongpoints to collect intelligence. These areas often do not have any connectivity with SIPR.

Opinions on the DCGS-A User Interface

The researcher sought to answer the last part of the research question by soliciting the participants' opinions on the DCGS-A UI with another series of questions. The survey asked the participants the dichotomous question "Is the DCGS-A user interface easy to use?" The researcher provided this question to the participants to seek an opinion on the overall ease-of-use of the DCGS-A UI. All fourteen (100 percent) participants indicated they did not know which version of DCGS-A they most recently used. Nine (64 percent) said the UI is not easy to use, while 4 (36 percent) answered "Yes" to the question. Table 10 identifies the participants' count and percent responses.

Table 10. Easy to Use DCGS-A Interface			
Question	Metric	Yes	No
Is the DCGS-A user interface easy to use?	Count	4	9
	Percent	31	69

Source: Created by author

Open-Ended Question: How could the DCGS-A user interface be improved?

The researcher provided participants with this open-ended question in attempt to reveal the participants' opinions about redesigning a more user-friendly version DCGS-A UI. Overwhelmingly, the responses from eight (57 percent) of the participants indicated an intuitive commercial-style UI integrated into DCGS-A would improve the DCGS-A UI. One of the participants simply stated:

[The] interface is difficult to use because I don't know how to create the products needed.

Open-Ended Question: What would make the DCGS-A user interface more functional?

This open-ended question provided the participants with an opportunity to share their thoughts on the ideas they had to improve the functionality of the DCGS-A UI. Two (14 percent) participants responded to the question. The theme of their responses indicated a need for better-trained users, and the integration of commercial-based software applications.

Research Question 3: Does the amount of time spent using and-or training on DCGS-A hinder the user's ability to operate the system?

Training on DCGS-A

The researcher provided a series of questions to the participants in attempt to identify training issues associated with the use of DCGS-A. The questions provided to the participants came in the form of dichotomous questions, multiple-choice questions, and open-ended questions.

The survey asked participants the question, “How much time have you spent using DCGS-A during the past year. One (7 percent) participant responded, indicating they had spent at least nine months but not more than a year using the system. Table 11 identifies the responses from the participant’s in count and percent format.

Table 11. Time Spent Using DCGS-A						
Question	Metric	None	Up to 2 weeks	At least 2 weeks but not more than 3 months	At least 3 but not more than 9 months	At least 9 months, but not more than a year
How much time have you spent using DCGS-A during the past year?	Count	13	0	0	0	1
	Percent	93	0	0	0	7

Source: Created by author.

The survey asked participants the question, “How much time have you spent training on DCGS-A during the past year?” Two (14 percent) participants responded, one (7 percent) stating they spent up to two weeks, while the other (7 percent) said they spent at least two but not more than four months training on the system. Table 12 depicts the participants’ count and percent responses.

Table 12. Time Spent Training on DCGS-A					
Question	Metric	None	Up to 2 weeks	Up to 2 months	At least 2 but not more than 4 months
How much time have you spent training on DCGS-A during the past year?	Count	12	1	0	1
	Percent	86	7	0	7

Source: Created by author.

The number of participants responding to these questions is not unusually low because the population that were provided the survey have been attending school at CGSC for the past seven months and have not had the opportunity to formally train on or use the system.

The researcher provided participants with the dichotomous question, “Is the amount of DCGS-A training you receive sufficient to provide you with the skills to operate the system?” Six (43 percent) participants answered “Yes”, while eight (57 percent) answered “No” to the question. The participant’s count and percent responses are depicted in Table 13.

Table 13. DCGS-A Training and Skill Retention			
Question	Metric	Yes	No
Is the amount of DCGS-A training you receive sufficient to provide you the skills to operate the system?	Count	6	8
	Percent	43	57

Source: Created by author.

Open-Ended Question: Comments on DCGS-A training

The researcher provided this open-ended comment box in attempt to capture any additional concerns the participants had on DCGS-A training. Two themes emerged from the comments that nine (64 percent) participants provided: training and ease of use. The training theme indicated training needs to be recurring and offered at CGSC as an elective. The second theme provided DCGS-A is not an intuitive system and training only provides an overview of “buttonology”. The participants’ comments provide an accurate account of the training concerns:

DCGS-A is a non-intuitive system that operators must use on a regular basis after training in order to retain the knowledge.

The training needs to be recurring, and it is ridiculous that CGSC does not offer DCGS-A training during AOC exercises or as an elective for the MI students attending the course.

Open-Ended Question: How is your ability to operate DCGS-A affected by the amount of time between uses?

The survey provided participants with an open-ended question to report how the length of time between uses affects the user's ability to operate DCGS-A.

Overwhelmingly, the theme of the eleven (79 percent) participants that responded indicated the more time spent not using DCGS-A affects a user's ability to operate the system because it is not easy to use. Some of the comments from the participants explained their frustration with the system:

DCGS-A operations are perishable skills that atrophy over time and required yearly refresher training to maintain currency with updates and retain skills previously learned.

[The ability] diminishes for two primary reasons: One, it is easy to forget the buttonology piece. Two, the system gets updated so the longer between uses, the more evolution of upgrades to catch up on.

DCGS-A simply doesn't have the capacity, the right tool, [nor] is it easy enough to use as a real warfighting system.

Survey Comments

The last page of the survey provided an open-ended comment box where participants could enter any final thoughts they had on the DCGS-A system. The research attempted to collect thoughts, ideas, or complaints about DCGS-A that the survey did not cover. Nine (64 percent) participants provided candid comments that identified a few areas of concern for the system. The following comments explain the challenges:

The worst part of DCGS-A is that it partially works in a mature theater with large bandwidth capabilities. It doesn't work as intended utilizing expeditionary equipment to power and connect it to the brain. In garrison, we utilize DCGS-A with shorepower (sic) and virtually unlimited bandwidth on a daily basis. This experience has caused commanders to set battle rhythms and personal expectations based on a functional DCGS-A. My experience has seen commanders consistently disappointed with intel personnel due to DCGS-A's limited performance in an expeditionary configuration.

FOUNDRY should just be a refresher. In addition, Soldiers should be using DCGS-A every day at their units. This would complement the RAF concept. There is no time for a relearn of DCGS...it should be done weekly at the bare minimum.

DCGS-A is a system that probably needs work, but what system doesn't. The MI Corps made DCGS-A their selling point so let's make it as good as possible.

DCGS-A is good for synchronization, but trying to do too much with just one system creates problems. Connectivity, speed, endurance, and compatibility are lacking in the system; this forces analysts to abandon it all together when the mission does not allow time to wait on those issues. The system is not compatible with the RAF concept as small teams of Soldiers operating in austere environments will not have the infrastructure to plug into the system.

The core theme that emerged from the comments indicate a need for DCGS-A to evolve into a system that can support the new Regionally Aligned Forces (RAF) concept. It appears that DCGS-A works well when operating the system in controlled environment, but placing that system into an environment where resources are constrained, the system does not function as intended. Additionally, training on DCGS-A seems to be a recurring theme. Many of the participants mentioned how training is good, but it is not enough to retain the skills needed to use DCGS-A with confidence over a prolonged frame of time.

Summary

Chapter 4 presented the results of the study. It included the qualitative and quantitative data provided through the survey instrument. In addition, it contained

statistical analyses of the quantitative data. An analysis of the overall survey instrument indicated several challenges facing the DCGS-A system that includes training and technology problems. The results of this study offers information pertinent to both the training plan and future development of the DCGS-A enterprise. Chapter 5 provides further in-depth analyses and explains the implications of the issues discovered in this chapter. Additionally, the researcher provides recommendations for DCGS-A developers, program managers, and the remainder of the U.S. Army intelligence community that are associated with the DCGS-A enterprise.

CHAPTER 5

CONCLUSION AND RECOMMENDATIONS

The Distributed Common Ground System-Army (DCGS-A) has received a lot of scrutiny from lawmakers, the media, and users. Top military leaders have proclaimed the system is capable of meeting the needs of the soldiers that use it, yet there are still negative connotations that the system is flawed. The aim of this study was to identify the inferred flaws by studying the affects the UI and the training plan has on the system's overall ease-of-use and functionality. The researcher collected data from a small fraction of the total amount of DCGS-A users from Army officers attending the Command and General Staff College (CGSC), resulting in low confidence that the results are representative of the whole DCGS-A user population.

The primary research question was, "How does the user interface within DCGS-A affect operability of the system?" This question was answered through multiple secondary questions to help understand what other issues affect the operability of DCGS-A, if the DCGS-A UI detracts from the capability to analyze and disseminate intelligence in a timely fashion, and if the amount of time spent using and-or training on DCGS-A hinders the user's ability to operate the system.

Attempting to discover issues affecting the operability of DCGS-A truly tested the researcher's ability to communicate the survey's intent clearly to the target audience. Although there were a relatively small number of individuals that participated in the survey, the information provided valuable data that the researcher translated into conclusions and recommendations.

Research Question 1: What other issues affect the operability of DCGS-A?

Discussion

In order to understand the reason why many perceive DCGS-A to not function as intended, a holistic study needs to identify the areas that challenge users of the system. A portion of this study was dedicated to identify the areas outside the DCGS-A UI that posed challenges to its users. Despite the fact that the small survey participant population does not necessarily represent the DCGS-A user population, the collected data possesses valuable feedback that leaders within the DCGS-A community can apply to improve the system.

DCGS-A Bandwidth Issues

The feedback provided from the survey indicates a need to provide more bandwidth in expeditionary settings. In expeditionary environments, the Trojan Spirit is supposed to support DCGS-A. According to the feedback, the Trojan Spirit cannot reliably provide enough bandwidth to support multiple DCGS-A fielded for any particular mission. In cases where there was enough bandwidth to support in training environments, the DCGS-A use in the field set a precedent and expectation for commanders. This expectation fails when units arrive to their expeditionary setting and DCGS-A fails to achieve its desired result and is unable to support the commander's intent.

DCGS-A Hardware Issues

Survey data indicated soldiers need to have the ability to easily service the DCGS-A. Normally, soldiers trained to operate DCGS-A do not service the hardware

because the Army has not trained them. DCGS-A is serviced by specially trained maintenance personnel that have the technical expertise to remove, replace, and-or repair the highly sensitive components of DCGS-A. The servicers come in two forms: contractors and soldiers. Both types appear in deployment and training environments. Units rely heavily on trained soldiers to fix DCGS-A when it breaks in expeditionary environments, while in garrison, and field training exercises. In some cases when essential components of DCGS-A break, the system could be rendered useless, at times leaving DCGS-A users without a machine to produce intelligence.

Recommendations

A portion of this study was dedicated to identifying issues other than the DCGS-A UI that affect the functionality and-or operability of the system. There is a true challenge to provide sufficient bandwidth to several DCGS-A systems operating in an expeditionary environment, and undoubtedly, that threshold will continue to be a challenge to meet as the U.S. military progresses into the future. Understanding the complexities of the bandwidth issues requires further in-depth analysis of the issues causing the problems. Future studies on this topic should focus on expanding the research to a wider DCGS-A user population, and identifying the root causes of the Trojan Spirit facilitating the perceived faulty communication of DCGS-A systems. Identifying solutions to the bandwidth problems in expeditionary environments are beyond the expertise of the researcher and should be a project taken on by the technical experts within the DCGS-A development field. The Army is heading towards aligning the force structure regionally. In order to mitigate hardware problems with a DCGS-A fielding in expeditionary environments, future studies should identify the feasibility of a DCGS-A

which is modular and adaptable to varying environments. These studies should include a focus to determine the practicality of a DCGS-A “lite” system that is an easily deployable system operating with a small footprint that has the basic tools necessary to accomplish expeditionary missions.

Research Question 2: Does the user interface of the current version of DCGS-A detract from the capability to analyze and disseminate intelligence in a timely fashion?

Discussion

Through the collection of survey data, the researcher attempted to understand which components of the current DCGS-A UI poses greatest challenges to system users. The tools within the DCGS-A UI are the operator’s link to creating, disseminating, analyzing, manipulating, and viewing intelligence. The DCGS-A UI has configured the tools to work independently, within specific groups interdependently throughout the system. Popular opinion indicated the system is not easy to use because of its compartmentalized systems approach. The survey provided participants with opportunities to elaborate their thoughts on an integrated workspace. Data from the survey indicated DCGS-A would be easier to use if its workspace were integrated. The opinions of the survey participants provided valuable input to this discussion.

DCGS-A Tools

The survey attempted to identify the tools that need some sort of improvement without delving into the specifics of what part of the tool that needs to be improved. The tools can be very powerful instruments when trained and used properly. The results of the Likert scale ratings on the DCGS-A tools indicate many of the survey participants had

little to no experience using many of the Ozone Widgets, Human Intelligence (HUMINT), and Full Motion Video (FMV) tools. Without significant feedback on these tools, the research cannot draw conclusions. The statistical analysis and feedback from the Likert scale ratings on the Geospatial, Collection Management, and All-source tools indicate a need to improve many of the Geospatial tools, and some of the Collection Management tools. The researcher concedes that the participants' responses to the Likert scale ratings of the DCGS-A tools do not provide an accurate reflection of the opinions of the entire DCGS-A user community because of the population's demographics, experience using the system, and survey "burn-out".

Workspace Integration

An integrated workspace would consist of combined tools, widgets, and programs where the tools and programs work within a single workspace area, and there is no workspace division between the programs. The majority of the survey participants said DCGS-A would be easier to use and more functional if the workspace was integrated. Some of the participants' comments compared the intuitive features of Palantir, Microsoft Office suite, and Google Earth to the researcher's vision of a DCGS-A with an integrated workspace. A DCGS-A UI to compliment an integrated workspace would be a challenge for DCGS-A program managers. It is uncertain that a redesigned DCGS-A workspace that has combined components would create a system that is easier to use. The need for a more functional and easy-to-use DCGS-A workspace was recommended by the survey participants however. Their responses indicate a need for the Army to redesign the DCGS-A workspace in order to make it more functional and easier to use.

Interoperability with other command systems such as the Command Post of the Future

(CPOF), and Blue Force Tracker (BFT) would make DCGS-A more functional and provide a more complete common intelligence picture.

DCGS-A User Interface Opinions

The majority of the participants responded the DCGS-A UI is not easy to use, indicating the system needs to be reformed. The survey participants also indicated the DCGS-A UI is not intuitive and the Army needs to redesign it. The opinion-based responses are indicative of the difficulties users have when operating on the system. The survey participants indicated the want for a system that is intuitive and easier to use based on popular commercial UI applications.

Recommendations

Although the survey was instrumental in providing valuable feedback to the researcher, it did not reach the entire anticipated audience. Feedback from the Likert scale ratings on the DCGS-A tools provided little value. Future studies on this topic should focus on expanding the population to the entire DCGS-A user population to include instructors at the U.S. Army Intelligence Center of Excellence (ICOE), technicians, and operators, in order to understand the areas of DCGS-A that need the most attention.

Based on the data obtained and analyzed in this survey, future studies should focus on the benefits and disadvantages of a redesigned DCGS-A which would include an integrated workspace. Topics within this research should identify the proper balancing of private sector partnerships against program goals and intuitive system design. Additionally, the Army should identify how to make the system more functional through

system interoperability. This portion of future studies should identify the integration of commercial-based software applications to improve functionality and ease-of-use.

It will not be easy for the Army to provide an easy-to-use robust product that offers that same capabilities as the current DCGS-A version. Future studies should identify how the Army can effectively utilize the cooperation, coordination, and the direction of private sector partners and contractors to design a robust product to offer better functionality, ease-of-use, and an intuitive DCGS-A UI. These studies should take into consideration the common characteristics and reasons why software fails presented in chapter 2 of this study.

Research Question #3: Does the amount of time spent
using and-or training on DCGS-A hinder the user's
ability to operate the system?

Discussion

In order to grasp the entire set of challenges facing the issues plaguing the DCGS-A UI, the researcher focused a portion of the study on the DCGS-A training program. An analysis of the training program indicated a robust program in place to provide every opportunity for users to understand how to use the system. The researcher concedes the survey participants do not represent the majority of DCGS-A users, and understands officers are not the primary every-day users that typically operate on the system. The DCGS-A community cannot discount the experience and feedback the survey participants provided however.

DCGS-A Training

The survey participants provided a mixed response when asked if the amount of training they received was sufficient to provide the skills they needed to operate DCGS-A. Sustainment and recurring training was a common theme from the survey data. According to the responses, system operators need more training to operate on the system more efficiently. The researcher does not ignore that everyday unit business may interfere with the ability to provide additional training to DCGS-A users. It is the leader's responsibility to ensure soldiers seize every training opportunity to ensure they receive the maximum amount of DCGS-A screen time. The survey participants also indicated DCGS-A training should be offered as an elective, or incorporated into the curriculum at CGSC. Just one of the survey participants said they had spent some time using and-or training on DCGS-A over the past year, which clearly indicates a need to provide training opportunities on DCGS-A at CGSC. Some of the survey participants indicated users must train on the system regularly because the system is not intuitive enough to simply pick up and use it whenever the mission requires its use. As the Army moves into an environment with decreased deployments and continued downsizing, it will become increasingly challenging for military intelligence professionals to integrate DCGS-A training time into their schedules.

Recommendations

It is apparent the DCGS-A training enterprise is a robust and rigorous program designed to support, maintain, retain, and educate soldiers on the use of DCGS-A. What is not apparent however, are the ties between training and skill retention. Further study should be directed towards the impact training has on the level of soldier skill retention.

This research project focused a small portion of the study on the opinions of mid-grade officers attending a PME course. This sample is clearly not representative of the DCGS-A user population. Future studies in this area should expand the population to include warrant officers, enlisted, and Department of the Army civilians that instruct on, and-or use DCGS-A. These studies should determine the effects a change in the DCGS-A training enterprise would have on the doctrine, organization, training, materiel, personnel, and facility (DOTMLPF) domains.

The hard approach is to create an intuitive system that has the same capacity as DCGS-A. An intuitive UI may decrease the amount of training users need to operate the system but further research needs to determine that feasibility. Survey participants constantly pointed to commercial products such as Microsoft Office, Google Earth, and Palantir as examples of intuitive systems that give users the ability to operate the systems with little to no training. The theme throughout the study indicates the system is not intuitive and soldiers must receive continual training to maintain their skills to operate DCGS-A. Survey participants compared DCGS-A to a language skill; it is a perishable skill unless it is used and maintained over time. Future studies in this area need to identify the relationship between skill retention through training, and skill retention using intuitive systems. These studies should include the impact an intuitive system would have on training and overall skill retention.

Further research should seek to identify specific areas of the DCGS-A training enterprise that do not provide the services required by system users. Additional research should focus on the positive and negative affects a change in the DCGS-A training enterprise would have on DOTMLPF when compared to a long-term cost associated with

the design, build, and fielding of an intuitive DCGS-A system. Studies should include a cost benefit analysis of adding to and-or taking away aspects of DCGS-A training.

Comments on DCGS-A

Discussion

The researcher provided an open-ended comment box at the end of the survey to provide a space for survey participants to present any additional thoughts on topics or areas of DCGS-A not covered in the survey. The researcher identified the core theme from the comments that indicated a need for DCGS-A to evolve into a system that can support the Regionally Aligned Forces (RAF) concept. The comments reinforced several key areas of DCGS-A which need to be improved. Some of the survey participants felt the Army should continue emphasizing DCGS-A training as a method of reinforcing the “buttonology” piece the system, while others felt the Army should scratch and rebuild the current system from the ground up. Some of the survey participants indicated the hardware and bandwidth issues present challenges to commanders attempting to utilize DCGS-A outside of garrison environments.

Projecting DCGS-A into the Future

According to the data collected from the survey, one of the biggest concerns with the current DCGS-A configuration is that it cannot support the future RAF concept. Some of the hardware and bandwidth issues elaborated earlier in this chapter indicate problems that will certainly evolve into larger problems when a conflict arises following a complete RAF implementation. Expeditionary environments present significant challenges for the current DCGS-A system configuration. Current problems such as those

identified in this study that are affecting DCGS-A bandwidth and hardware will be amplified in expeditionary environments because of environmental constraints, and adverse and austere conditions.

Recommendations

Future studies on this topic should expand research to include a wider population of DCGS-A users to gather data on issues affecting DCGS-A hardware and bandwidth. The research should seek to identify the key problems affecting the system hardware and bandwidth, and how it will affect deployment in an expeditionary environment when used in a RAF concept. After identifying the key issues affecting DCGS-A hardware and bandwidth, the Army should develop a long-term fix. Along with the training recommendations provided in this chapter, future studies should focus on if the RAF concept will affect DCGS-A system deployment. Further research in this area should identify the challenges and solutions that austere environments have on DCGS-A.

Summary and Implications

Chapter 5 presented conclusions and recommendations to the DCGS-A community. The research sought to identify the core issues plaguing the DCGS-A UI and attempted to analyze the structure and effectiveness of the training program. Mainly due to time constraints, the researcher was unable to expand the study's population to a wider DCGS-A population, which limited the data received. This limited population is not necessarily a representative DCGS-A user population, however, the data extracted from the survey is valuable and represents common themes plaguing DCGS-A.

The conclusions indicate a need to alleviate perceived problems with the DCGS-A hardware and bandwidth, a more user-friendly and easy to use UI, reforming the DCGS-A training plan, and providing a system that is compatible with the RAF concept. The disjointed conglomerate of tools and widgets that compose the DCGS-A UI complicates the system and forces the soldier to depend on recurring training to retain their skills. Army leaders need to decide if they are going to continue down the road with a robust DCGS-A training concept, or completely redesign the DCGS-A UI into an intuitive system that does not require much training to sustain user skills. The suggestions provided in this study are a starting point for decision-makers. It is important the individuals involved in making the big decisions for DCGS-A to understand the implications of failing to continue studying the issues affecting the system and failing to make positive changes to system will have on future generations of intelligence warfighters.

GLOSSARY

Distributed Common Ground System–Army: The Distributed Common Ground System – Army is the Army’s Intelligence component that gathers intelligence spanning all echelons from space to mud. DCGS-A enables decision makers to save lives right now by gathering, analyzing and sharing intelligence information into a common system. This enterprise utilizes intelligence information and open source technology (such as Amazon, Google and Twitter) to create a scalable environment for collaboration and intelligence production. Overall, it has 600 sources of information.

Private sector: The area of the nation’s economy under private rather than governmental control.

Public sector: The public sector consists of governments and all publicly controlled or publicly funded agencies, enterprises, and other entities that deliver public programs, goods, or services

User interface: A graphical display of information displayed on a computer screen based on metaphors of documents, files, folders, and applications that are used for various forms of manipulation.

APPENDIX A

ONLINE SURVEY

Defense Common Ground System - Army

The purpose of this survey is collect information from individuals who use the Defense Common Ground System - Army (DCGS-A). It also satisfies the research requirement for my Masters of Military Arts and Science (MMAS) degree.

Your participation is voluntary and all results are confidential.

The survey should take approximately 10-15 minutes to complete.

For questions or concerns regarding the survey, contact:
Dr. Maria Clark, Human Protection Administrator
maria.l.clark.civ@mail.mil

This survey has been approved by the CGSC Quality Assurance Office. The survey control number is 15-03-032.

Page 1 ▾

Next

Save



POWERED BY
 ALLEGIANCE

Have you been trained to use the
Distributed Common Ground
System-Army (DCGS-A) system?

☐ Yes

☒ No


Page 2 ▾

Back

Next

Save



POWERED BY
 ALLEGIANCE

Have you been trained to use the Distributed Common Ground System-Army (DCGS-A) system?

☒ Yes

☐ No

In what component do you serve?

☐ Regular Army

☒ Army Reserves

☐ Army National guard

Are you?

☐ Enlisted

☒ Warrant Officer

☐ Commissioned Officer

What is your military occupation specialty?

☐ 35D - All-Source Intelligence Officer

☒ 35E - CI/HUMINT Officer

☐ 35F - HUMINT Intelligence Officer

☐ 35F - Intelligence Analyst

☐ 35G - Signals Intelligence/Electronic Warfare Officer

☒ 35G - GEOINT Imagery Analyst

☐ 35L - CI Agent

☐ 35M - HUMINT Collector

☐ 35N - SIGINT Analyst

☒ 35P - Cryptologic Linguist

☐ 35S - Signals Collector/Analyst

☒ 35T - Intelligence and EW Systems Maintainer/Integrator

☐ 35X - Chief Intelligence Sergeant

☒ 35Y - Chief CI/HUMINT Intelligence Sergeant

☐ 35Z - SIGINT Senior Sergeant/SIGINT Chief

☒ 350F - All-Source Intelligence Technician

☐ 350G - Imagery Intelligence Technician

☒ 351L - CI technician

☐ 351M - HUMINT Collection Technician

☒ 352N - Traffic Analyst Technician

☐ 353T - Intelligence Systems Integration/Maintenance Technician

Page 2

Back

Next

Save



POWERED BY
ALLEGIANCE

Do you have experience using DCGS-A during times other than training?

☐ Yes

☒ No

How long have you been using DCGS-A?	Less than 1 year	At least 1 but less than 3 years	At least 3 but less than 5 years	More than 5 years
Time spent using DCGS-A	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

How much time have you spent using DCGS-A (other than for training) over the past year?	None	Up to 2 weeks	At least 2 weeks but not more than 3 months	At least 3 but not more than 9 months	At least 9 months, but not more than a year
Time spent using DCGS-A	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

Page 3 Back Next Save POWERED BY ALLEGIANCE

Is the amount of DCGS-A training you receive sufficient to provide you the skills to operate the system?

☐ Yes

☒ No

How much time have you spent training on DCGS-A over the past year?	None	Up to 2 weeks	Up to 2 months	At least 2 but not more than 4 months
Time spent training on DCGS-A	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

Comments on DCGS-A training:

Page 4 Back Next Save POWERED BY ALLEGIANCE

Have you deployed within the past 3 years?

☐ Yes

☒ No

Did you use DCGS-A during your most recent deployment?

☐ Yes

☒ No

How much time did you spend using DCGS-A on your most recent deployment?	None	Up to two months	At least 2 but not more than 6 months	At least 6 but not more than 12 months
Time using DCGS-A	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

How is your ability to operate DCGS-A affected by the amount of time between uses?

Page 5

Back

Next

Save



POWERED BY
ALLEGIANCE

DCGS-A workspace

An integrated DCGS-A workspace is defined as a workspace where tools and programs work with one another, programs are combined and run within a singular workspace area, and there is no workspace division between the programs.

Would an integrated DCGS-A workspace make DCGS-A easier to use?

☐ Yes

☒ No

What combined components of DCGS-A would make DCGS-A easier to use?

Operating DCGS-A

An integrated DCGS-A workspace is defined as a workspace where tools and programs work with one another, programs are combined and run within a singular workspace area, and there is no workspace division between the programs.

Would an integrated DCGS-A workspace make DCGS-A more *functional*?

☐ Yes

☒ No

What combined components of DCGS-A would make DCGS-A more *functional*?




Page 7 ▾

Back

Next

Save



POWERED BY
 ALLEGIANCE

DCGS-A bandwidth

Do bandwidth problems affect user functionality of DCGS-A?

☐ Yes

☒ No

Do hardware (i.e. servers, computers, etc...) problems affect user functionality of DCGS-A?

☐ Yes

☒ No


Page 8

Back

Next

Save



POWERED BY
 ALLEGIANCE

DCGS-A user interface

What version of DCGS-A do you currently use/train on?

- ☐ v 3.15
- ☒ v 3.16 increment 1, release 1
- ☐ v 3.16 increment 1, release 2
- ☐ I don't know

Is the DCGS-A user interface easy to use?

- ☐ Yes
- ☒ No

How could the DCGS-A user interface be improved?

What would make the DCGS-A user interface more functional?

Page 9

Back

Next

Save



POWERED BY
ALLEGIANCE

Geospatial Tools	Needs substantial improvement	Needs some improvement	No improvement needed	No experience with tool
3d Fly Thru	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
ArcGIS 10	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Geospatial Analysis	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Map services and product dissemination	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

Collection Management	Needs substantial improvement	Needs some improvement	No improvement needed	No experience with tool
ISR sync matrix	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Mission planning/management	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Requirements management	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
RFI	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Roles, users, units, and privileges setup	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
User management	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

Full Motion Video	Needs substantial improvement	Needs some improvement	No improvement needed	No experience with tool
MIScreener	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
MISnapReporter	<input checked="" type="radio"/>	<input checked="" type="radio"/>	<input checked="" type="radio"/>	<input checked="" type="radio"/>
Socet GXP	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

HUMINT	Needs substantial improvement	Needs some improvement	No improvement needed	No experience with tool
Break down PIRs	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Export to HOT-R	<input checked="" type="radio"/>	<input checked="" type="radio"/>	<input checked="" type="radio"/>	<input checked="" type="radio"/>
HUMINT collection requirement (sphere of influence) source visualization	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Unit task order	<input checked="" type="radio"/>	<input checked="" type="radio"/>	<input checked="" type="radio"/>	<input checked="" type="radio"/>
Unit status	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

Ozone Widgets	Needs substantial improvement	Needs some improvement	No improvement needed	No experience with tool
ATHEN-A report widget	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
ATHEN-A message widget	<input checked="" type="radio"/>	<input checked="" type="radio"/>	<input checked="" type="radio"/>	<input checked="" type="radio"/>
ATHEN-A status widget	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
BAT dossier widget	<input checked="" type="radio"/>	<input checked="" type="radio"/>	<input checked="" type="radio"/>	<input checked="" type="radio"/>
BAT search widget	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Cascading query widget	<input checked="" type="radio"/>	<input checked="" type="radio"/>	<input checked="" type="radio"/>	<input checked="" type="radio"/>
Common map widget	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
DIB query widget	<input checked="" type="radio"/>	<input checked="" type="radio"/>	<input checked="" type="radio"/>	<input checked="" type="radio"/>
DIMS-F dossier widget	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
DIMS-F search widget	<input checked="" type="radio"/>	<input checked="" type="radio"/>	<input checked="" type="radio"/>	<input checked="" type="radio"/>
GeoDiscover	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
GeoPackage	<input checked="" type="radio"/>	<input checked="" type="radio"/>	<input checked="" type="radio"/>	<input checked="" type="radio"/>
GeoPrint	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
GeoWebPrint	<input checked="" type="radio"/>	<input checked="" type="radio"/>	<input checked="" type="radio"/>	<input checked="" type="radio"/>
GeoWebView	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
IWEDA widget	<input checked="" type="radio"/>	<input checked="" type="radio"/>	<input checked="" type="radio"/>	<input checked="" type="radio"/>
Weather overlays widget	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Weather report widget	<input checked="" type="radio"/>	<input checked="" type="radio"/>	<input checked="" type="radio"/>	<input checked="" type="radio"/>

I appreciate your willingness to participate in this survey, however, this research is focused on individuals with specific experience.

Thank you for your time.

Please close your browser to end the survey.

Defense Common Ground System - Army

Comments:

**Thank you for taking the time to complete this survey.
Your support is appreciated.**

Please click the **Finish button to submit your responses.**

Page 15 ▾	Back	Finish	Save		POWERED BY  ALLEGIANCE
-----------	------	--------	------	--	--

BIBLIOGRAPHY

- 34th Infantry Division. *After Action Report, Predeployment* vol 1 (July 2010).
- Akiki, Pierre, Arosha K. Bandara, and Yijun Yu. "Adaptive Model-Drive User Interface Development Systems." *ACM Computing Surveys* 47, no. 1 (April 2014): 9-1 to 9-33.
- Anderson, Christian U., and Soren Pold. *Interface Criticism: Aesthetics Beyond Buttons*. Oakville, CT: The David Brown Book Company, 2011.
- Association for Computing Machinery. *Proceedings of the 2010 ACM SIGCHI Symposium on Engineering Interactive Computing Systems*. New York, NY: ACM Order Department, 2010.
- Boehm, Barry W. "An Experiment in Small Scale Application Software Engineering." *IEEE Transactions on Software Engineering* SE-7, no. 5 (September 1981): 482-494.
- Brooks Jr., Frederick P. "The Mythical Man-Month: After 20 Years." *IEEE Software* 12, no. 5 (September 1995): 57-60.
- Brown, A. W., S. Iyengar, and S. Johnston. "A Rational Approach to Model-Driven Development." *IBM Systems Journal* 45, no. 3 (July 2006): 463-480.
- Capaccio, Tony. "Senators to Probe Air Force's #1 Billion Failed Software." Bloomberg.com. January 24, 2013. Accessed November 1, 2014. <http://www.bloomberg.com/news/2013-01-24/senators-to-proble-air-force-s-1-billion-failed-software.html>.
- Charette, Robert N. "Why Software Fails." *IEEE Spectrum*. September 5, 2005. Accessed October 20, 2014. <http://spectrum.ieee.org/computing/software/why-software-fails>.
- Dictionary.com. "Private Sector." Accessed October 8, 2014. [http://dictionary.reference.com/browse/private sector](http://dictionary.reference.com/browse/private%20sector).
- Dilanian, Ken. "Army Leaders Defend Flawed Intelligence System DCGS-A." *The Charleston Gazette*. July 10, 2014. Accessed January 30, 2015. <http://search.proquest.com/docview/1544209181?accountid=28992>.
- Distributed Common Ground System-Army. "About DCGS-A." Last modified May 7, 2014. Accessed October 8, 2014. http://dcgsa.apg.army.mil/about_dcgsa/.
- Dube, Sikhungo, and Daniela Danescu. "Supplemental Guidance: Public Sector Definition." The Institute of Internal Auditors. December 2011. Accessed August

- 29, 2014. [https:// na.theia.org/standards-guidance/Public%20Documents/Public%20Sector%20Definition.pdf](https://na.theia.org/standards-guidance/Public%20Documents/Public%20Sector%20Definition.pdf).
- Ewusi-Mensah, Kwaku. *Software Development Failures*. Cambridge, MA: The MIT Press, 2003.
- Findlater, Leah, and Krzysztof Z. Gajos. "Design Space and Evaluation Challenges of Adaptive Graphical User Interfaces." *AI Magazine* 30, no. 4 (Winter 2009): 68-73.
- Fink, Arlene. *How to Conduct Surveys: A Step-by-Step Guide*. Thousand Oaks, CA: Sage Publications, 2009.
- Flatten, Mark. "Army Can't Track Spending on \$4.3b System to Track Spending, IG Finds." *Washington Examiner*, September 4, 2014. Accessed November 1, 2014. <http://www.washingtonexaminer.com/army-cant-track-spending-on-4.3b-system-to-track-spending-ig-finds/article/2552808>.
- Flynn, Michael to Deputy Commanding General for Support, USFOR-A, July 2, 2010, in Memorandum.
- Fowler Jr., Floyd J. *Survey Research Methods*. Thousand Oaks, CA: Sage Publications, 2009.
- Gallagher, Sean, and Larry Barrett. "McDonald's: McBusted." *PCMag.com*. July 2, 2003. Accessed November 3, 2014. <http://www.pcmag.com/article2/0,2817,1173624,00.asp>.
- Gregor, MAJ Robert. "Intelligence, Intelligence Trends." *Bronco Team Newsletter: Tactics, Techniques, and Procedures*, no. 10-21 (January 2010): 1-71.
- Katelinin, Victor, and Mary Czerwinski. *Beyond the Desktop Metaphor: Designing Integrated Digital Work Environments*. Cambridge, MA: The MIT Press, 2007.
- "Key to Successful Software: Teamwork." *Resource* 6, no. 2 (February 1999): 4.
- Khurana, Anil, and Stephen R. Rosenthal. "Integrating the Fuzzy Front End of New Product Development." *Sloan Management Review* 38, no. 2 (Winter 1997): 103-120.
- Lakhampal, B. "Understanding the Factors Influencing the Performance of Software Development Groups: An Exploratory Group-Level Analysis." *Information and Software Technology* 35, no. 8 (August 1993): 468-473.
- McConnell, Steve. *Software Estimation: Demystifying the Black Art*. Redmond, WA: Microsoft Press, 2006.

- McGarry, Brendan. "Army Units in Afghanistan Slam Intel System." *Military.com*. February 6, 2014. Accessed September 14, 2014. <http://www.dodbuzz.com/2014/02/06/army-units-in-afghanistan-slam-intel-system>.
- McGregor, John D., Dirk Muthig, Kentaro Yoshimura, and Paul Jensen. "Guest Editors' Introduction: Successful Software Product Line Practices." *IEEE Software* 27, no. 3 (May 2010): 16-21.
- Meixner, Gerritt, Fabio Paterno, and Jean Vanderdonckt. "Past, Present, and Future of Model-Based User Interface Development." *I-com* 10, no. 3 (March 2011): 2-11.
- Meixner, Gerritt, Gaelle Calvary, and Joelle Coutaz. "Introduction to Model-Based User Interfaces." W3C, January 7, 2014. Accessed October 31, 2014. <http://www.w3.org/TR/mbui-intro/>.
- Nelson, R. Ryan. "IT Project Management: Infamous Failure, Classic Mistakes, and Best Practices." *MIS Quarterly Executive* 6, no. 2 (June 2007): 67-78.
- Raskin, Jef. *The Humane Interface; New Directions for Designing Interactive Systems*. New York, NY: ACM Press, 2000.
- Righi, Carol, and Janice James. *User-Centered Design Stories: Real-World UCD Case Files*. San Francisco, CA: Morgan Kaufmann Publishers, 2007.
- Scott, John, Fritz Schultz, and Dan Risacher. "Open Technology Development (OTD): Lessons Learned and Best Practices for Military Software." Accessed September 12, 2014. <http://mil-oss.org/resources/books/open-technology-development>.
- Seale, Clive, Giampietro Gobo, Jaber F. Gubrium, and David Silverman. *Qualitative Research Practice*. Thousand Oaks, CA: Sage Publications, 2007.
- Sommerville, Ian. "Software Process Models." *ACM Computing Surveys* 28, no. 1 (March 1996): 269-271.
- Stross, Randall. "Billion Dollar Flop: Air Force Stumbles on Software Plan." *The New York Times*, December 8, 2012. Accessed November 1, 2014. <http://www.nytimes.com/2012/12/09/technology/air-force-stumbles-over-software-modernization-project.html>.
- Sue, Valarie M., and Lois A. Ritter. *Conducting Online Surveys*. Thousand Oaks, CA: Sage Publications, 2012.
- United States Army Intelligence Center of Excellence-Military Intelligence School. "Distributed Common Ground System-Army V3, Version 1.0." November 2013. Accessed January 3, 2015. <https://rdl.train.army.mil/catalog-ws/view/100.ATSC/155E7214-A5B8-4870-B16B-4102FE998330-1383762781051/doc.pdf>.

US Congress, House. *Testimony before the Subcommittee on Oversight, Committee on Ways and Means*. 108th Cong., 2nd sess. February 12, 2004.

———. *Testimony by Secretary Penetta and Gen. Dempsey Before the House Armed Services Committee*. 112th Cong., 1st sess, October 13, 2011.

Wong, Kristina. “Army Defends Intel System Ahead of House Hearing.” TheHill.com. March 25, 2014. Accessed January 27, 2015. <https://thehill.com/policy/defense/201604-army-defends-intel-program-ahead-of-house-hearing>.

Wright, Austin. “The Army’s Multibillion Dollar ‘Money Pit.’” Politico.com. May 29, 2013. Accessed September 14, 2014. <http://dyn.politico.com/printstory.cfm?uuld=83921989-537E-4565-A508-33BF3D02CB06>.